

Nr. 3/84/4,80 DM/40 ÖS/SFR 4,80/Dezember/Januar

SUPER KASSETTEN ANGEBOT!

ALLE LISTINGS NUR 10 DM

EIGENBAU: 80 KILOBYTE FÜR DIE TI-KONSOLE

30 SEITEN LISTINGS FÜR IHREN TI 99/4A

SOFTWARE IM TEST: GRAPHIK-TABLETT

ALLE PINS AUF EINEN BLICK

SECHS NÜTZLICHE UNTER-PROGRAMME

TIPS & TRICKS

Grüß Gott – Gruezi – Guten Tag

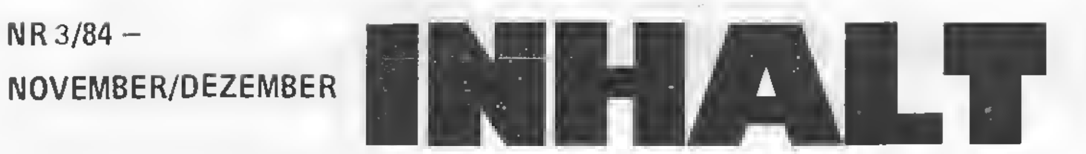
Hier ist sie, die dritte Ausgabe der TI REVUE, schon sehnlichst erwartet.
Allen Anrufern noch einmal zur Kenntnis:
Wir erscheinen nur alle zwei Monate, und zwar in den ungeraden!
Keine Ausgabe ohne Probleme.
Diesmal durch die Einsender von Programmen.
Bitte, verpackt die Datenträger gut, am besten in Polsterbeutel.
Viele Kassetten kamen nicht an, weil sie nur in einem simplen Briefumschlag steckten, wir erhielten die leeren Hüllen.
Und irgendwann diverse Kassetten ohne Alles.
Auch der Kassettenversand klappt immer noch nicht richtig, obwohl wir uns alle Mühe geben.
Nur: Wir kopieren in einem Großbetrieb und da gelten halt mal 100 000 Musik-Cassetten für Udo Jürgens vor, wenn wir mit unseren 200 oder 300 Bestellungen kommen.
Aber nur so können wir diesen Preis halten.
Ein Hinweis für die, die Diskettenwünsche äußerten:
Bitte, kopiert Euch die Programme von der Kassette selbst, denn es gibt leider noch keine Diskettenkopiermaschinen. Das müßte, wie bei Euch, von Hand erfolgen, Stück für Stück. Programm für Programm.
Wer soll das bezahlen?

Und zum Schluß noch eine gute Nachricht:
Ab dem nächsten Heft werden wir ein „heißes Telefon" einrichten. Unter einer ganz bestimmten Nummer werden Euch dann Experten Eure Fragen beantworten.
Also: Nächstes Heft nicht versäumen!
Es kommt Ende Januar 1985.

Bis dahin die besten Grüße zu den Festtagen und zum Jahreswechsel

Euer TI REVUE Team
Und nicht vergessen: Seid Ihr mit dem Heft unzufrieden, sagt es uns.
Seid Ihr zufrieden, sagt es weiter.

NR 3/84 –
NOVEMBER/DEZEMBER



IN DIESEM HEFT LESEN SIE

Impressum

TI-REVUE, die Zeitschrift für den TI PC und den TI 99/4A erscheint zweimonatlich im TI-Aktuell-Verlag, Postfach 1107, 8044 Lohhof. Redaktion: Senator-Presse-Service. Verantwortlich für den Inhalt: Klaus Herdin. Verantwortlich für Anzeigen: Bruno Redase. Verlagsbetreuung: München Aktuell-Verlag. Alle: Postfach 1107, 8044 Lohhof. Anfragen bitte nur schriftlich. Vertrieb: Verlagsunion Wiesbaden. Mitarbeiter dieser Ausgabe: SPS, das Computer-Aktuell-Team, Pieter Coates, Hartmut König, Günther Stahl, Hans J. Adler, Dirk Gripp, Alfred Nietschke, Armand Marx, Berth. Wölfelschneider, N.+L. Keller, Jorg Wolters, Rainer Ziegler. Es gilt die Honorarliste des Verlages. Für unaufgefordert eingesandte Manuskripte und Listings keine Haftung. Bei Einsendung von Texten, Fotos und Programmträgern erteilt der Autor dem Verlag die Genehmigung für einen einmaligen Abdruck, für den eventuellen Datenträger-Versand werden gesonderte Vereinbarungen getroffen.

Erstverkaufstag dieser Ausgabe: 28.11.1984
Es gilt Preisliste Nr.1 vom 1.7.1984.
Media-Unterlagen bitte anfordern bei Bruno Redase, co Aktuell-Verlag, Postfach 1107, 8044 Lohhof.
Für telefonische Anfragen: 089/28 60 35
Printed in Germany



FRAGEN & ANTWORTEN

VON TEXAS ENTTÄUSCHT

Als Besitzer des Assemblers sowie des XBasics bin ich enttäuscht darüber, daß TEXAS INSTRUMENTS nicht in der Lage ist, die Software des TI-99/4A aufeinander abzustimmen. Es gelingt mir nämlich nicht, irgendein Assemblerprogramm mit den Befehlen:
CALL INIT
CALL LOAD ("DSK1. IRGENDWAS")
vom XBasic aus zu laden. Das Diskettenlaufwerk springt zwar an, aber das XBasic gibt nach ca. 2 Sekunden die Fehlermeldung *UNRECOGNIZED CHARACTER* aus und bricht den Ladevorgang ab.
Auf die Anfrage bei TEXAS INSTRUMENTS, was es denn damit auf sich habe, stieß ich nur auf Unwissenheit (wie so oft – seufz!).
Meine Frage ist also: Wie kann ich mein XBasicmodul dazu bringen, ein Assemblerprogramm anzunehmen?

Stephan Jeremias
Nerotal 12
6200 Wiesbaden

*Die Funktion des Extended Basic Moduls ist vollkommen in Ordnung. Die Meldung *UNRECOGNIZED CHARACTER* ist eine Schutzfunktion des Assembler Moduls gegenüber Extended Basic. Grundsätzlich ist es nämlich möglich, jedes Assembler Programm mit X-Basic zu laden, wenn es nur beim assembeln NICHT mit der Option C=Compressed versehen wurde. Ist diese, zugegeben auf der Diskette platzsparende, Version verwendet worden, kann dieses Programm nur noch über den Assembler mittels LOAD AND RUN gestartet werden.*

KABEL-PROBLEME

Seit Ende Februar '84 bin ich Besitzer eines TI-99/4A und bin eigentlich auch ganz zufrieden.
Doch eines Tages lief meine kleine Schwester durch den Raum und stolperte dabei über das Antennenkabel meines TIs. Seitdem habe ich fast jedesmal, wenn das Kabel leicht angestoßen wird oder „falsch" liegt, einen Systemausfall. Statt des gerade eingeladenen oder eingetippten Programms habe ich nur noch Funkstille, manchmal auch einen Pfeifton, und ein graues Fernsehbild. Zwar schaffe ich es zumeist durch Drehen oder „Bugsieren" (Veränderung der Lage) auf das Testbild zurückzukommen, aber der Speicher ist gelöscht, und damit auch das Programm!
Wie kann ich die Störung beseitigen und wieviel würde eine Reparatur kosten?

Heinz-Josef Eckers
Hermann-Hesse-Str. 7
4050 Mönchengladbach 1

Offensichtlich hat Ihre kleine Schwester das Kabel des Pal-Modulators derart beschädigt, daß bei einer bestimmten Bewegung ein Kurzschluß verursacht wird. Ein einfacher Kabelriß würde nämlich „nur" das Bild verschwinden lassen, ohne Systemabsturz. Am besten senden Sie dieses Teil mit Kaufnachweis zu Texas Instruments nach Freising, noch während Ihrer einjährigen Garantiezeit ein. Desweiteren fehlt Ihnen bei Ihrer Konsole die Gebrauchsanleitung. Sie haben jedoch schon ein Buch Basic/Extended Basic für Anfänger erworben. Dieses beinhaltet auch die komplette Gebrauchsqnweisung, genauso wie die komplette deutsche Anleitung für das Extended Basic Modul. Übrigens, Extended Basic Module gibt es im einschlägigen Fachhandel in ausreichenden Mengen, wenn auch zu entsprechenden Preisen. Sehen Sie doch mal im Anzeigenteil nach.

DISKMANAGER DEFEKT?

Seit etwa 2 Jahren besitze ich den TI 99/4A mit Erweiterungsbox, Controller-Laufwerk, Schnittstelle, RAM-Erweiterung, 32 K-Ex-Basic, Minimemory, Editor-Assembler und vieles mehr. Nun zu meinem Problem: das Diskettenmanagermodull initialisiert nicht mehr ordnungsgemäß. Beim Initialisieren (Bereiche werden in Zahlen angezeigt) geht er bis 17, dann rattert das Laufwerk ein wenig und dann macht es weiter. Dieses Spiel wiederholt sich in anderen Bereichen etwa 40mal. Bei der Fertigmeldung zeigt er: 316 Bereiche frei 40 belegt.
Bei der späteren Benutzung dieser Disketten werden die Programme nicht einwandfrei gespeichert und werden auch nicht mehr eingelesen.
Nun habe ich einen Wunsch und zwar: eventuelle Tips oder Hilfe oder mögliche Ursache dieses Fehlers oder gibt es ein Programm, womit man mit Ex-Basic initialisieren kann!

Peter Hoops
Skagerrakstr. 10c
2960 Aurich 1

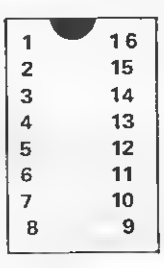
Ihr Diskettenmanagermodul ist offensichtlich defekt. Wir empfehlen Ihnen, das Modul an Texas Instruments zur Reparatur einzusenden. Eine Möglichkeit, Basic heraus zu initialisieren, ist uns nicht bekannt.

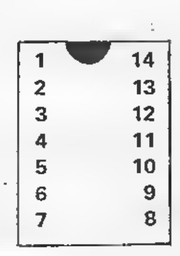
WO SIND DIE BEINCHEN

Nr. 2 gesehen, gekauft. Schnell durchgeblättert wie immer, Stop – auf Seite Nr. 8 „Achtkanal-Ein- und Ausgabe".
Ein Spitzen-Bericht und dazu noch mit einem Schaltplan.
Ich habe am nächsten Tag die Bestellung der angegebenen Chips gemacht. Drei Tage danach kam die Bestellung ins Haus.
Lötkolben aufgeheizt, Chips 741s85 schon in der Hand und, na eben und wie kommt er nun hin???
16 Beine und 14 Anschlüsse nach Schaltplan.
Erster Gedankenblitz: Sind die Beine numeriert? Ergebnis negativ.
Mit blasser Miene untersuche in dic anderen Schwarz-Beiner.
Resultat: Enttäuschung.
Für einen Nicht-Elektroniker sind die Schaltungen zwar sehr anschaulich gezeichnet, nur die Praxis ist eben doch einiges komplexer.
Nun heute ist der 2.10.84, bei mir zuhause ist mein alter TI 99, daneben das Minimem und darauf die 7 Waisenkinder, die nach den passenden Werkschuhen suchen.

Mario Roshardt
Mettlerstr. 4
8640 Rapperswil/ Schweiz

Nachdem diese Probleme bereits mehrfach aufgetreten sind, haben wir uns entschlossen, die grundsätzliche Pinbelegung, die bei allen integrierten Schaltkreisen gleich ist, darzustellen. Zur Verdeutlichung dient nachstehendes Bild, mit dem hoffentlich alle Unklarheiten beseitigt werden können.





Anschlüsse von oben betrachtet

FRAGEN & ANTWORTEN

DA-WANDLER FÜR DEN TI

Ich will meinen TI 99/4A zur Steuerung eines Schrittmotors verwenden. Wie kann ich dazu einen D/A-Wandler anschließen? Kann man eine universelle Ein/Ausgabekarte anschließen?
Edwin Halbleib
Herrengasse 12
8781 Motten

Die in Heft 2 veröffentlichte Schaltung bietet die Möglichkeit, einen D/A-Wandler anzusteuern, wobei nur die Ausgangsleitungen des 74 LS 259 mit den Eingängen des D/A-Bausteins verbunden werden müssen.
Verwenden kann man z.B. den A/D-Wandler TL 507, der auch als D/A-Baustein eingesetzt wird. Die notwendige Software sowie der Hardwareanschluß wird in einem der nächsten Hefte veröffentlicht.

MEIN DRUCKER DRUCKT ZU BREIT

An meiner Anlage habe ich einen „Epson RX 80"-Drucker angeschlossen, der jedoch beim Listen volle 80 Zeichen ausdruckt. Somit ist mir nicht mehr die Möglichkeit gegeben, daß ich das ausgeworfene Listing vernünftig abheften kann, da mir einfach kein Heftrand mehr bleibt.
Meine Frage: Gibt es eine Routine, womit ich dem Drucker den Befehl geben kann, nicht mehr als 40 (oder je nach Belieben) Zeichen auszudrucken.
Udo W. Jobsky
Vorsterhauser Weg 34
4700 Hamm 1

Mit den Befehlen

```
10 OPEN #1: "RS232..."
20 PRINT #1:CHR$(27);
   CHR$(n);
30 PRINT #1:CHR$(27);
   "Q";CHR$(n);
40 CLOSE #1
```

wird der linke und rechte Rand für den Drucker RX-80 festgelegt.
Diese Routine kann entweder ins Programm eingebaut oder als separate Unterroutine gestartet werden. Die Randbegrenzung bleibt solange im Drucker gespeichert, bis dieser entweder ausgeschaltet oder CHR$(64) (RESET) gesendet wird.

SCHWACHSINNIGE IDIOTEN-REDAKTEURE

Spätestens auf Seite 5 Ihres ersten Heftes wurde mir voller Entsetzen klar, daß es sich wieder einmal um genau die selben schwachsinnigen Tips handelt, wie sie an anderer Stelle schon hunderte von Malen erschienen waren.
Was diesen „Tips" jedoch die Krone aufsetzt und gleichzeitig auch der eigentliche Grund meines Schreibens ist, ist der Artikel auf Seite 6 Ihrer zweiten Ausgabe,der mit der nahezu sarkastischen Überschrift versehen ist, die da lautet „elegante(!) Befehle ohne X"
Was hier dem Leser geboten wird, grenzt (einmal davon abgesehen, daß es ohnehin schon hundertmal anderswo zu lesen war) an reine Unverschämtheit!
Doch möchte ich meine Kritik gerne fundieren: Beginnen möchte ich mit dem ersten „Tip": Da konnte man doch tatsächlich lesen, daß man Pi abrufen kann, indem man es zuvor als Variable eingibt. Wenn Sie etwas derartiges als Tip auffassen, dann möchte ich Ihnen doch einmal raten "IDIOTEN§="TI-Revue-Redakteure"" einzugeben und dann zu schauen was passiert, wenn man IDIOTEN§ abruft.
Ach, man kann übrigens Pi abrufen indem man 4*ATN(1) eingibt. Aber pi=3.141...ist vermutlich leichter zu behalten.
Bevor ich bei der Kritik dieses „Tips" noch weiter ins Bösartige abrutsche, möchte ich mich dem nächsten zuwenden: Natürlich ist auch dieser „Bartaufwickelmaschinen"-Trick vollkommen schwachsinnig!
Einmal davon abgesehen, daß die eckigen Klammern nicht akzeptiert werden, daß außerdem in Zeile 110 ein Seg§ zuviel steht und daß im zweiten Seg§-Befehl die Länge-Parameter fehlt, also abgesehen von diesen Kleinigkeiten, die jede für sich das Funktionieren der Subroutine unmöglich machen, ist diese Routine auch noch so entsetzlich unrentabel, daß man Speicherplatz spart, wenn man auf sie verzichtet!
Natürlich ist mir klar, daß Sie das oben gesagte nicht verstanden haben (wenn doch: Herzlichen Glückwunsch; Idiotentest bestanden), darum will ich es noch einmal näher erläutern: Wenn man zu der Subroutine springen will, muß man zunächst drei Variabeln definieren. Das macht zusammen mit der Gosub-Zeile 4 Zeilen. Benutzt man die Routine jedoch jedesmal gleich an Ort und Stelle und setzt die Variabeln sofort ein, dann fällt die Return-Zeile weg und es läßt sich nachweisen, daß die übrigen drei Zeilen weniger Speicherplatz benötigen als die vorherigen vier Zeilen.
Wolfgang Müskens
Kranenburger Str. 224
4180 Goch 6

Wir Idioten verneigen uns in tiefer Ehrfurcht vor dem Wissen dieses Lesers.

TI STREIKT

Beim Laden eines Programmes von Kassette auf den TI, mit angeschlossener Peripheriebox (32K-Erweiterung) und dem Extended Basicmodul, da streikt mein TI. Er nimmt das Programm nicht auf. Ziehe ich nun das Extended Basicmodul heraus und klemme die Peripheriebox ab, dann klappt das Laden. Da ich aber fast alle Programme auf Diskette habe, möchte ich auch die wenigen restlichen Kassettenprogramme auf die Diskette bringen.
Uwe Brandauer

In den meisten Fällen läßt sich das geschilderte Problem im Direktmodus durch den Befehl Call-Files (1) New beseitigen. Dieser Befehl schafft im Speicher einige hundert Bit Platz, so daß in den meisten Fällen die Kassettenprogramme geladen werden können. (Die Peripherieerweiterung benötigt einige Bit Platz im Konsolenspeicher, der dann bei den besagten Basicprogrammen fehlt.)
Wenn auch diese Methode versagt, bleibt Ihnen nichts anderes übrig, als mit abgeklemmter Peripheriebox Ihr Kassettenprogramm zu laden, die Hälfte davon zu löschen, diesen Teil auf die Kassette zurückzuladen, mit dem zweiten Teil genauso zu verfahren, anschließend Extended-Basic und Box wieder anzuklemmen, hineinzuladen, auf Diskette zur bringen und die beiden Teile wieder zusammenzumergen.
Dann müßte es gehen. Lediglich bei Kassettenprogrammen, die überlang und mit Protected gesichert sind, versagen sämtliche Methoden. Hier gibt es im Moment leider noch keinen Weg. Aber vielleicht weiß hier ein anderer Leser Rat.

BILDSCHIRM WEG

Wenn ich etwas länger am TI eingebe, kam es schon öfters vor, daß der Bildschirm auf einmal weg war, und die Speichererweiterung sich von selbst ausschaltete. Außerdem hatte ich schon Schwierigkeiten mit dem Load-Programm. Beim Einschalten müßte der Computer das Programm, das unter Load

»

FRAGEN & ANTWORTEN

abgespeichert wurde, einlesen. Oftmals hat der Computer dieses Load nicht gesucht. Es brummte dann auf dem Bildschirm oder der TI zeichnete chaotische Bilder. Seit einiger Zeit habe ich die 32 K-Karte in einen anderen Port vertauscht und bis jetzt läuft er wieder. An was kann das liegen?
Uwe Brandauer

So wie Sie das Problem schildern, sind es eindeutige Kontaktprobleme. Wir empfehlen Ihnen, sämtliche Kontakte am Modulport, am Peripherieport sowie in der Box vorsichtig mit einem Elektronikreinigungsmittel zu reinigen. Dann müßte es eigentlich wieder gehen.

Zum Thema MBI-Interface der Firma Klein in Wuppertal erreichte uns ein Schreiben einer Wuppertaler Anwaltskanzlei.
Darin wird gebeten, darauf hinzuweisen, daß es sich bei der in der Ausgabe 2 von TI-REVUE genannten Firma Klein nicht um ihren Klienten, die Firma Hans Klein, Elektronikbauteile Großhandlung, Postfach 200702, Wuppertal 2, handele.
Die Redaktion hat sich von der Richtigkeit dieser Feststellung überzeugt und bedauert, daß durch eine zufällige Namensgleichheit hier ein Unternehmen in einen falschen Verdacht geriet.

DAS MBI, DIE FIRMA KLEIN UND DIE ERFAHRUNGEN UNSERER LESER

Ich besitze seit einiger Zeit das MBI-Interface TI 99/4A auf Centronics. Für normales Drucken ist es zu gebrauchen, aber sobald man eigene Zeichen oder Grafik ausdrucken will, klappt nichts mehr. Nach einer Rückfrage bei MBI erklärt man mir, daß man für Grafikdruck die eingebauten Schalter auf 8 Bit umstellen muß. Dies klappte auch nach mehrmaligen Versuchen nicht.
Auch die in der Anleitung beschriebene Umstellung der Baud-Rate wird nicht mit Erfolg belohnt.
Ein großer Mangel ist, daß das Interface nicht abgeschirmt ist. Dies macht sich durch einen verschneiten Bildschirm bemerkbar. Die Zuleitung vom Interface zum Drucker muß mit Alufolie oder anderen geeigneten Materialien abgeschirmt werden.
H.-Günter Stahl
4100 Duisburg 26

*

Mir ging es ganz ähnlich wie diesem Herrn (W.M.), jedoch habe ich mit der Fa. Klein keine schlechte Erfahrung gemacht.

Auch ich wollte einen Drucker anschließen und war auf der Suche nach einer Möglichkeit. Im Metro-Großmarkt erfuhr ich zum ersten Mal von einem Multi Board Interface für die Silver-Reed EX 42, abgestimmt auf TI 99/4A.
Die Freude nun drucken zu können war groß, denn ich entschloß mich, diese beiden Geräte, Schönschreibdrucker und Multi Board Interface der Fa. Klein, zu kaufen.
Der erste Versuch, den Drucker in Gang zu bringen, schlug fehl.
Ein Anruf bei der Fa. Klein in Wuppertal brachte mir die Information, wie dieses Interface eingestellt sein muß.
Ich tat so, wie mir gesagt wurde, und mein Drucker funktioniert einwandfrei.
Werner Hassler

*

Der Grund für meinen Brief ist der arme Mathias E., der, wie Sie in der letzten Ausgabe schrieben, arge Probleme mit seinem Drucker bzw. Interface hat. Ich besitze ebenfalls einen MBI und den Seikosha GP 550A. Da ich auch 2 Stunden lang Probleme mit dem Interface hatte, hoffe ich, daß meine Lösung auch dem M.E. hilft.
Zuerst, bei dem MBI handelt es sich um ein serielles Interface, d.h. die Einstellung der Datenlänge ist sehr wichtig.
Der Tip ist, daß M.E. die Datei mit OPEN#1: "RS232.BA=4800.DA=8" eröffnet, wobei DA=8 bedeutet, daß 8 Datenbits kommen. Falls sich jetzt überhaupt nichts mehr drucken läßt, müßte das Interface geöffnet werden und die DIL-Schalter 3&4 auf die Stellung OFF&OFF gebracht werden, und danach geht es nochmal wie oben.
Ralf Biedermann
Ortfeld

*

Da ich ebenfalls im Besitz eines MBI-Multiboard-Interface der Firma Klein bin, erging es mir am Anfang ähnlich wie Herrn Sendelbach.
Trotz Probieren aller Möglichkeiten, entlockte ich meinem Seikosha Drucker GP100-A kein einziges Zeichen. Irgendwann gelang es dann doch. Hier meine Lösung:
Multiboard aufschrauben.
DIL-Schalter auf folgende Stellung bringen:
SCHALTER 1 ON
SCHALTER 2 OFF
SCHALTER 3 ON
SCHALTER 4 ON
SCHALTER 5 ON
SCHALTER 6 OFF
SCHALTER 7 ON
SCHALTER 8 ON
Drucker eröffnen mit

```
10 OPEN #1:"RS232.
   BA=4800.DA=8"
20 PRINT #1:.........
```

Und schon konnte es losgehen; sowohl bei der Grafik "CHR$(8)", als auch beim normalen Drucken geht es ohne Probleme.

*

Bezüglich Ihres Berichtes über die angeblichen Praktiken der Firma Computer-Klein, bin ich nicht ganz Ihrer Ansicht. Daß Sie diesen Brief, trotz der Zeilennummern, lesen können, verdanke ich dem Multiboard Interface, Hersteller: Fa. Klein! Ich habe im Februar 1984 über meinen Büromaterial-Großhändler hier in Frankfurt ein Multi-Interface zum Preis von ca. DM 650,– bestellt und bekam dieses ca. vier Wochen später geliefert. Da ich meine Anlage beruflich beim Kernkraftwerksbau in Brokdorf einsetzen mußte, benötigte ich auch einen vernünftigen Drucker. Diesen besorgte ich mir über den Otto-Versand, bekam aber das passende Interface dazu nicht mitgeliefert. Lediglich über die Niederlassung des Druckerherstellers bekam ich dann die Anschrift des Interface-Herstellers. Trotz eines Fehlers im Interface, den ich erst vor vier Wochen beheben lassen konnte – eine Lötstelle war verkehrt angelötet.– bin ich mit dem Interface sehr zufrieden. Die Reparatur erfolgte kostenlos innerhalb von vier Tagen auf Garantie! Ihrem Herrn W.M. kann ich nur empfehlen, sich nochmals an die Fa. Klein zu wenden; er sollte aber möglichst schreiben: z. Hd. v. H. König – er ist der maßgebende Techniker dort.
Josef Bachmaier



TIPS & TRICKS

SINGLE STEP FÜR DEN TI 99/4A

Manche Assemblerfreaks werden sicher schon bemerkt haben, daß im Debugger ein Befehl für Singlestep implementiert wurde.
Leider fehlt jedoch die entsprechende Hardware. Diesem Übel möchte ich mit meiner Schaltung abhelfen.
Nachdem ich ca. 40 Seiten Listing produziert hatte, hielt ich den kompletten Debugger Source Code in den Händen. Der entsprechende Teil für Single-Step war schnell gefunden. Texas Instruments wollte die notwendige Hardware vielleicht einmal als Option anbieten, hat das aber leider nicht getan.
Wofür kann man eigentlich Single-Step verwenden?
Diese Frage läßt sich relativ leicht beantworten. Nehmen wir ein langes fehlerhaftes Assemblerprogramm und testen die Funktionsweise mit Hilfe des Debuggers. Bisher war es nur möglich, das Programm nach dem Befehl E (EXECUTE) anzuhalten, durch vorheriges Setzen von Breakpoints (Unterbrechungsstellen).
Wollte man nach jedem Programmbefehl eine Überprüfung der Register- und Speicherinhalte vornehmen, so konnte dies nur mit dem zahllosen Setzen von Breakpoints realisiert werden.
Mit meiner Hardwareerweiterung, die sich übrigens problemlos in den Speech-Synthesizer implementieren läßt, wird diese umständliche Fehlersuchmethode restlos beseitigt.
Die Funktionsweise der Schaltung ergab sich zwangsweise durch die bereits vorhandene Software, die ich nicht ändern wollte.
Das S-Kommando gibt somit nach jedem ausgeführten Befehl die Kontrolle an den Debugger zurück, man kann nach jeder Anweisung des Benutzerprogramms eine Überprüfung durchführen.
Die Anweisung S 6 <ENTER> veranlaßt den Debugger, 6 Programmschritte abzuarbeiten, erst danach wird die Kontrolle wieder an den Debugger übergeben.
Kommen wir nun zur Hardwarebeschreibung:
Wie aus dem Schaltplan erkennbar, erfolgt eine Dekodierung der Adresse 13C0, die oberhalb der RS232-1-Schnittstelle liegt.

Die Notwendigkeit zur Dekodierung dieser Adresse ergab sich aus der Softwaregegebenheit des Debuggers.
Auszug aus der Debugger-Software:

```
          -
          -
          INC STARTA
TRACE1    MOV §T9900,
          §FLAG
          SBO 0
          RTWP
*    LOAD ENTRY
LOAD
LD        EQU $
          -
          -
```

Nachdem der Befehl SBO 0 erfolgte, müssen noch zwei Befehle ausgeführt werden. Erstens der Rücksprung des Debuggers von der Single-Step-Routine zum Benutzerprogramm (RTWP) und zweitens der nächste anstehende Befehl des zu überprüfenden Assemblerprogramms. Glücklicherweise zeigt der TMS 9900 die Befehlsausführung über die Steuerleitung IAQ an, wenn zwei IAW-Zyklen aufgetreten sind, wurden auch zwei Befehle abgearbeitet.
Solange die CRU-Adresse 13C0 nicht angesprochen wird, erfolgt ein Sperren der Leitung A15/CRUOUT, wodurch der Ausgang des Gatters und somit die Presetleitung der 74ALS74 auf logisch 1 (=5V) liegt.
Die dauernde Aktivität der IAQ-Steuerleitung führt dazu, daß der 0V-Pegel der D-Leitung auf den Ausgang des Flipflops (Q) durchgetaktet wird.

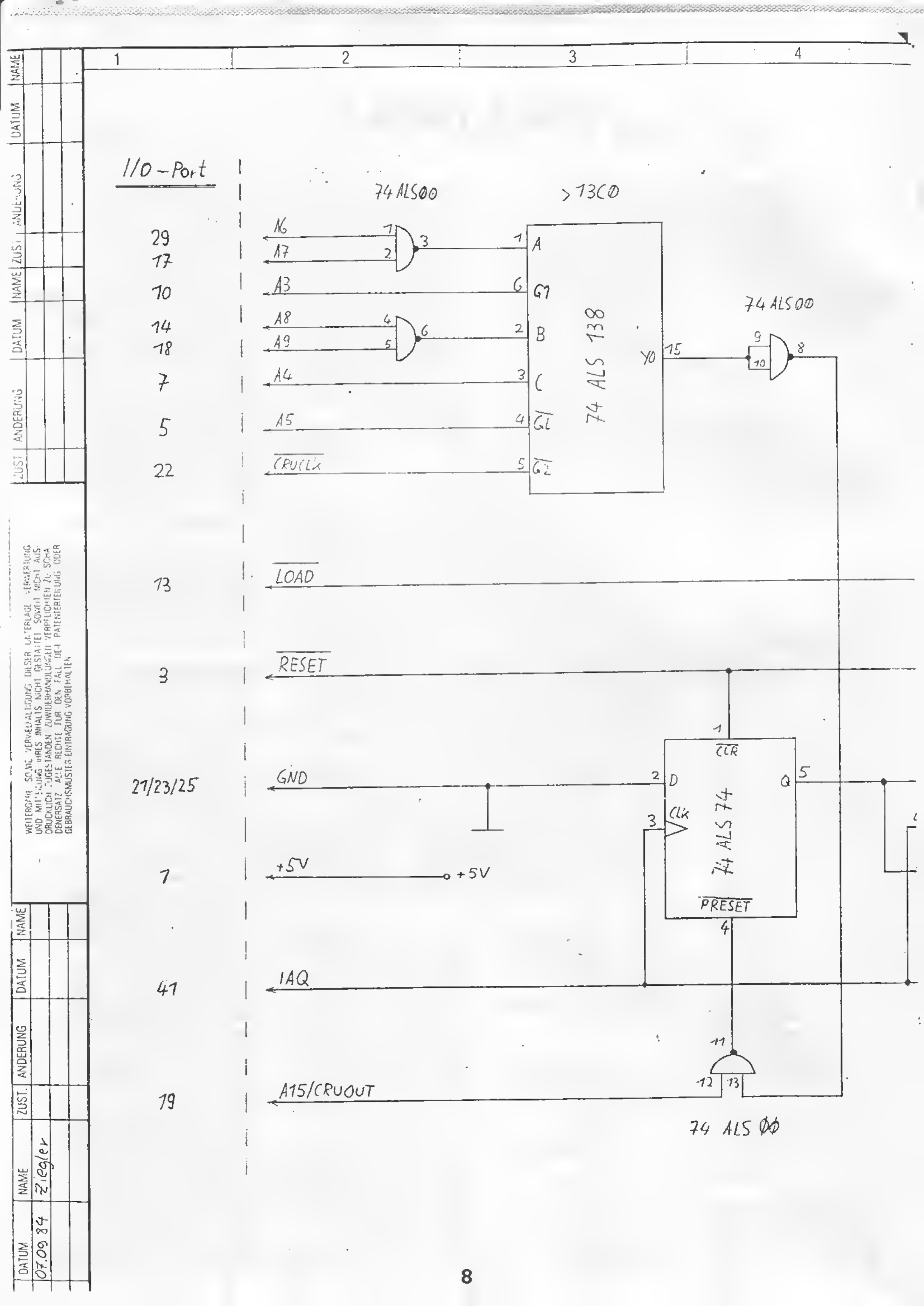
Diese logische 0 (=0V) wird von den zwei Flipflops 74ALS109 ebenfalls bis zum Ausgang durchgeschoben. Bedingt durch den Load-Interupt, der bei logisch 0 aktiv wird, erfolgt der Abgriff am invertierten Q-Ausgang des Flipflops (im Ruhestand logisch 1).
Bei Verwendung des S-Kommandos wird die Leitung A15/CRUOUT freigegeben und der Ausgang des 74ALS74 auf 1 gesetzt. Diese 1 bewirkt das Anlaufen der nachfolgenden Flipflops, die nunmehr nach zwei IAQ-Zyklen einen Load-Interrupt erzeugen.
Tritt ein Load-Interrupt auf, so erfolgt ein Sprung zur Memory Adresse FFFE. Der Inhalt dieser Speicherzelle wird als Programmfortsetzadresse verwendet, beim Inhalt B000 wird ab B000 mit der Fortführung des Assemblerprogramms begonnen. Das Laden der Fortsetzadresse wird vom Debugger beim Start vorgenommen.
Durch diese Zwangsunterbrechung erhält der Benutzer nach jeder S-Anweisung die Kontrolle über sein Programm zurück.
Zum Schluß möchte ich noch näher auf die Verwendung von ALS (Advanced Lowpower Schottky) eingehen. Der Grund liegt in der Stromersparnis, die rund 30% betragen dürfte. Der Einsatz dieser ICs findet volle Anerkennung bei der ohnehin ziemlich überlasteten Spannungsversorgung der Konsole. Eine Verwendung von LS-Bausteinen liegt aber auch im Rahmen des möglichen, zumal dies für den Geldbeutel weniger strapaziös sein dürfte.
Ein Anschluß der Schaltung am Grom-Port scheidet von vornherein aus, da wichtige Signale (IAQ, LOAD) völlig fehlen.
Ich hoffe, mit dieser Schaltung allen Assembleranwendern ein nützliches Werkzeug zum Testen Ihrer Programme gegeben zu haben.

Reiner Ziegler

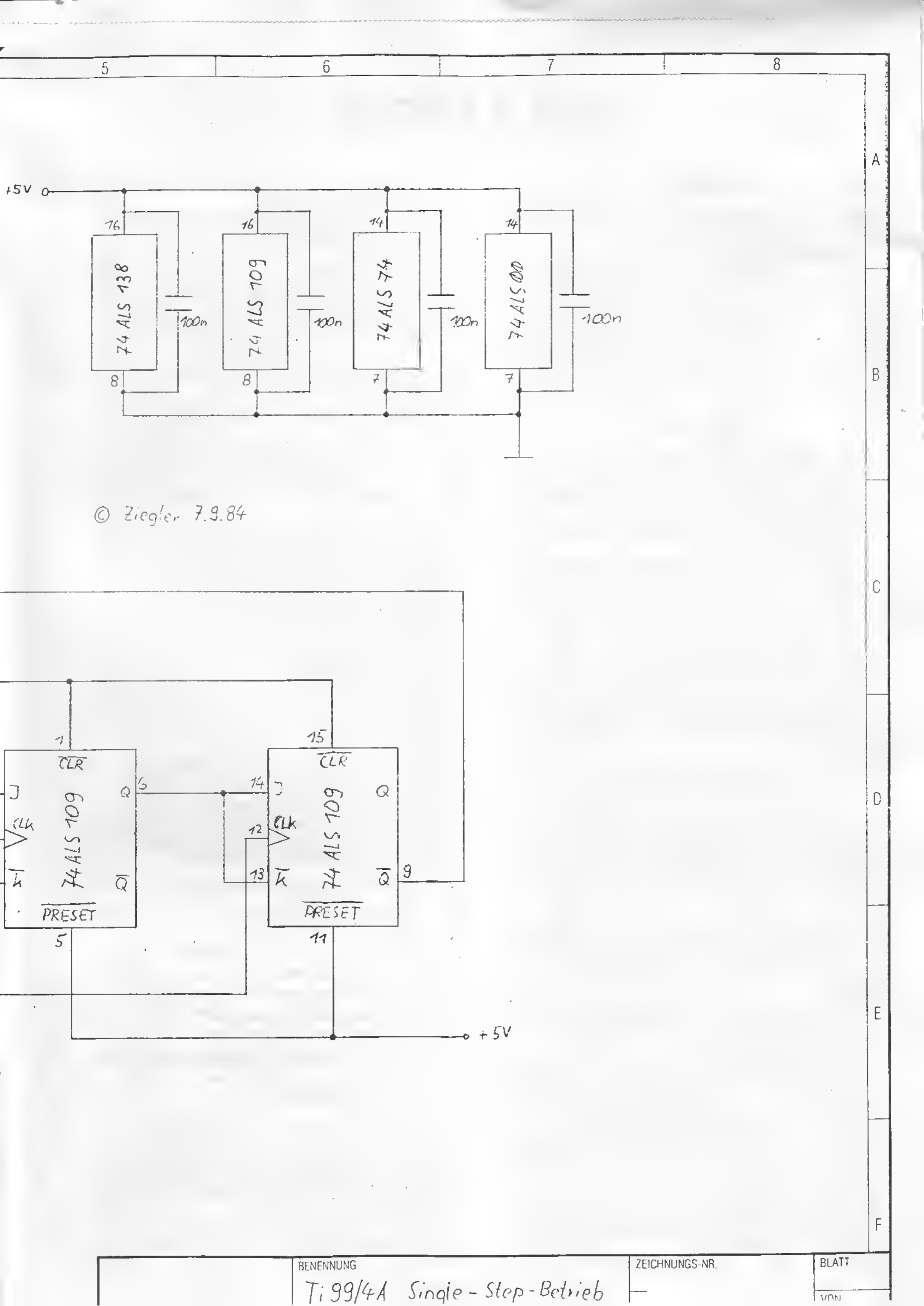
NÜTZLICHES WERKZEUG ZUM TESTEN ALLER PROGRAMME FÜR ALLE ASSEMBLER-ANWENDER

UNSER SOFT-WARE-TEST IN HEFT 2

Im letzten Heft haben wir eine ganze Reihe von neuen Spielmodulen aus den USA getestet. Bei sämtlichen Modulen handelt es sich um Original-TI-Software, die bisher am deutschen Markt nicht erhältlich war. Uns haben daraufhin sehr viele Anfragen nach Bezugsquellen erreicht. Diese Spiele sind bisher am deutschen Markt nur sehr beschränkt erhältlich.
Wie wir aus den Aufträgen unserer Anzeigenkundschaft ersehen können, werden die Spiele voraussichtlich im Weihnachtsgeschäft in ausreichender Menge zur Verfügung stehen. Wer sie kaufen möchte, wendet sich am besten an unsere Anzeigenkunden. Der Verlag selbst handelt nicht mit Software und Hardware; auch ist es uns leider nicht möglich, derartige Anfragen weiterzuleiten. Zu den Preisen möchten wir noch bemerken, daß es sich um die umgerechneten Dollarpreise bezogen auf den Monat August dieses Jahres handelte. Zu diesen Preisen kommt selbstverständlich noch der geänderte Wechselkurs sowie Fracht und Zoll hinzu.
Dadurch werden diese Spiele auf dem deutschen Markt um ca. die Hälfte teurer sein als in den USA. Dies liegt aber nicht an den Händlern, sondern einzig und allein an Fracht und Zollkosten.



I/O-Port
74 ALS00
>13C0
74 ALS 138
74 ALS00
74 ALS 74
74 ALS 00
A6
A7
A3
A8
A9
A4
A5
CRUCLK
LOAD
RESET
GND
+5V
IAQ
A15/CRUOUT
CLR
PRESET
Ziegler
07.09.84


5
6
7
8
A
B
C
D
E
F
+5V
74 ALS 138
74 ALS 109
74 ALS 74
74 ALS 00
100n
16
8
14
7
© Ziegler 7.9.84
CLR
PRESET
J
CLK
K
Q
Q̄
1
5
15
14
12
13
9
11
+5V
BENENNUNG
Ti 99/4A Single - Step - Betrieb
ZEICHNUNGS-NR.
BLATT
VON

TIPS & TRICKS

UNTERPROGRAMME

Anhand der nun folgenden Beispiele will ich zeigen, wie einfach und nützlich das Erstellen von Unterprogrammen mit dem TI-99/4A ist.
Bevor ich jedoch mit den eigentlichen Unterprogrammen beginne, ein kurzer Überblick über die Gesetzmäßigkeiten, welche unbedingt beim Gebrauch von solchen Unterprogrammen bekannt sein müssen, um lästige Fehlermeldungen zu vermeiden. Der TI mit EXTENDED BASIC besitzt insgesamt 3 spezielle Befehle, die zur Erstellung und zum Gebrauch von Unterprogrammen notwendig sind.
Die Befehle sind: SUB ( )
SUBEND
SUBEXIT
Durch "SUB(Programmname(Variable))" kann man einem Unterprogramm einen Namen zuordnen, außerdem können Variablen und Strings hinzugefügt werden, wodurch eine allgemeine Anwendung möglich wird.
In einem Programm kann das z.B. so aussehen:
>100 SUB TEST (X,Y,A$).
Test ist der Name des Unterprogramms und dahinter stehen die Variablen. Möchte man das Unterprogramm aufrufen, dann muß man z.B. schreiben: >10 CALL TEST (5,7,"TSET").
Der zweite Befehl SUBEND gibt an, wann das Unterprogramm zuende ist und weist den Computer an, zur Ausgangsadresse zurückzukehren.
Möchte man jedoch das Unterprogramm früher verlassen, kann man das nicht einfach mit GOTO erreichen, sondern man muß einen extra dafür vorgesehenen Befehl verwenden, und zwar SUBEXIT.
Der Computer verhält sich dabei genauso wie bei SUBEND.
Ein häufiger Fehler, der am Anfang gemacht wird, ist der, daß man hinter die Unterprogramme noch Befehle wie FOR TO ; NEXT ; DATA etc. stehen hat. Man erhält dann eine Fehlermeldung: "ILLEGAL AFTER SUBPROGRAM". Man darf also hinter Unterprogramme keine Befehle setzen, lediglich END und REM sind erlaubt. Unterprogramme selbst können natürlich, solange der Speicher noch Platz hat, beliebig oft hintereinander geschrieben werden.
Die Anwendung von Unterprogrammen kann für den Besitzer einer Diskettenstation bei der Erstellung von Programmen besonders nützlich sein, denn der TI-99/4A hat eine sogenannte MERGE Funktion.
Die MERGE Funktion ermöglicht einem nämlich, Unterprogramme zu laden, ohne daß dabei ein bereits im Speicher befindliches Programm gelöscht wird.
Aufpassen muß man da nur mit den Adressen. Hat das im Speicher befindliche Basic Programm nämlich die gleiche Adresse belegt wie das Unterprogramm, das man „mergen" möchte, dann wird diese Zeile vom Unterprogramm überschrieben.
Um dieses ungewollte Überschreiben möglichst zu vermeiden, haben die folgenden Unterprogramme Adressen über 30000.

Mit Unterprogrammen zu arbeiten und damit Programme zu entwerfen, ist sehr zeitsparend und es lohnt sich, eine kleine Sammlung solcher Unterprogramme anzulegen.

Pieter Coates

```
30015 SUB BLINK(YPOS,XPOS,SP
EED,TIME,SATZ$)
30016 IF YPOS<=0 OR XPOS<=0
OR SPEED<0 THEN SUBEXIT
30017 FOR TI=1 TO TIME :: DI
SPLAY AT(YPOS,XPOS)SIZE(LEN(
SATZ$)):SATZ$
30018 FOR T=1 TO SPEED :: NE
XT T
30019 DISPLAY AT(YPOS,XPOS)S
IZE(LEN(SATZ$)):""
30020 FOR T=1 TO SPEED :: NE
XT T
30021 CALL KEY(0,K,S):: IF S
<>0 THEN SUBEXIT
30022 NEXT TI :: SUBEND
```

Mit dem BLINK Unterprogramm kann man Texte blinken lassen. Sei es, um auf sie besonders aufmerksam zu machen oder nur um den äußerlichen Eindruck eines Programms zu verbessern.
Das Unterprogramm verlangt fünf Variable, die man angeben muß. Durch die ersten zwei "YPOS,XPOS" muß man die Anfangsposition des Satzes festlegen. Durch "SPEED" muß man die Blinkfrequenz eingeben. Zusätzlich muß dann noch die Gesamtzahl der Blinkvorgänge eingegeben werden; man kann jedoch jederzeit durch Drücken einer beliebigen Taste den Blinkvorgang unterbrechen. In dem String "SATZ$" soll der Text definiert sein.
Angewendet könnte das dann so aussehen:
>100 CALL BLINK(23,2,20,100, "IRGEND EINE TASTE DRUECKEN")
Die ersten beiden Zahlen, 23 und 2, geben die Koordinaten des ersten Buchstaben des Satzes an, 20 ist die Blinkfrequenz und 100 schließlich ist die Anzahl, wie oft der Satz blinken soll.

```
30009 SUB STEXT(Z,W$)
30010 IF LEN(W$)>30 THEN SUB
EXIT
30011 Q=INT(32-LEN(W$))/2+1
30012 FOR AX=Q TO LEN(W$)+Q-
1 :: S$=SEG$(W$,AX-Q+1,1)::
FOR AZ=31 TO AX STEP -1
30013 CALL HCHAR(Z,AZ+1,32):
: CALL HCHAR(Z,AZ,ASC(S$))::
 NEXT AZ :: NEXT AX
30014 SUBEND
```

Das "STEXT" Unterprogramm ermöglicht auch das Schreiben eines Textes auf dem Bildschirm, jedoch wird hier ein Text in seine Bestandteile, also in seine Buchstaben, zerlegt und diese werden dann Stück für Stück vom linken Bildrand in die Mitte geschoben.
Man muß dabei nur die Zeile eingeben, in die der Text geschrieben werden soll, und den Text selber.

TIPS & TRICKS

In einem Programm kann das so aussehen:
>100 CALL STEXT(10,"SCHIEBE TEXT")
Hier wird in Zeile 10 der Text "SCHIEBE TEXT" nach und nach auf den Bildschirm geholt.

```
30000 SUB CTEXT(Y,SND,A$)
30001 IF LEN(A$)>32 THEN SUB
EXIT
30002 XP=INT(16-LEN(A$)/2)
30003 FOR ST=1 TO LEN(A$)
30004 CALL HCHAR(Y,XP+ST,30)
:: CALL HCHAR(Y,XP+ST,ASC(SE
G$(A$,ST,1)))
30005 IF SND<110 THEN 30007
30006 FOR L=0 TO 30 STEP 10
:: CALL SOUND(-100,SND,L)::
NEXT L
30007 NEXT ST
30008 SUBEND
```

Bei diesem Programm kann man einen Text durch einen Cursor schreiben lassen. In der Klammer befinden sich drei Variablen, wobei die erste "y" angibt, in welcher Zeile der Cursor den Text schreiben soll.
Durch "SND" kann man die Frequenz des Tones eingeben, der dann stets bei jedem Buchstaben ertönt. Möchte man jedoch keinen Ton haben, so braucht man die Variable "SND" nur gleich null zu setzen.
Der Text selbst wird durch den String AS repräsentiert.
Möchte man das Unterprogramm aufrufen, kann man schreiben:
>100 CALL CTEXT (5,220," BELIEBIGER TEXT")
Hier wird der Cursor angewiesen, in Zeile 5 mit einem Ton von 220 Hz den Text "BELIEBIGER TEXT" zu schreiben.

```
30030 SUB HC(GERAET$)
30031 OPEN #10:GERAET$,OUTPU
T
30032 FOR Y=1 TO 24 :: Q$=""
30033 FOR X=32 TO 1 STEP -1
:: CALL GCHAR(Y,X,WE):: WT$=
CHR$(WE):: Q$=WT$+Q$
30034 NEXT X
30035 PRINT #10:TAB(22);Q$ :
: NEXT Y :: CLOSE #10
30036 SUBEND
```

HC steht für Hardcopy, wodurch die Aufgabe dieses Unterprogrammes angedeutet wird. Dieses Programm ermöglicht es ASCII Zeichen, die auf dem Bildschirm sind, direkt zum Drucker zu übertragen, jedoch keine selbstdefinierten Symbole. Man muß lediglich das Gerät angeben, das den Drucker steuert, entweder seriell, dann muß man schreiben RS232.BA=....., oder parallel, dann heißt es PIO.

Das Programm ist sehr einfach aufgebaut. Der Copiervorgang läuft daher recht schnell ab.

```
30023 SUB RAHMEN(Z1,SP1,Z2,S
P2)
30024 CALL CHAR(91,"000000FF
FF",92,"1818181818181818",93
,"0000001F1F181818",94,"0000
00F8F8181818",95,"1818181F1F
",96,"181818F8F8")
30025 CALL HCHAR(Z1,SP1,93):
: CALL HCHAR(Z1,SP2,94)
30026 CALL HCHAR(Z2,SP1,95):
: CALL HCHAR(Z2,SP2,96):: CA
LL HCHAR(Z1,SP1+1,91,SP2-SP1
-1)
30027 CALL HCHAR(Z2,SP1+1,91
,SP2-SP1-1):: CALL VCHAR(Z1+
1,SP1,92,Z2-Z1-1)
30028 CALL VCHAR(Z1+1,SP2,92
,Z2-Z1-1)
30029 SUBEND
```

Dieses Unterprogramm ermöglicht es einen Rahmen auf dem Bildschirm herzustellen, um beispielsweise eine Überschrift von einem Menü hervorzuheben. Man muß lediglich 4 Koordinaten angeben, und zwar die der oberen linken Ecke (Z1,SP!) und die der unteren rechten Ecke (Z2,SP2).
Das Programm belegt allerdings 6 Zeichen (ASCII Code 91–96) und kodiert diese um. Will man diese Zeichen jedoch im Original später verwenden, so muß man die Definition rückgängig machen, z.B. mit CALL CHARSET.
Möchte man also ein Rechteck, mit der 2. Zeile und der 2. Spalte als obere linke Ecke und der 15. Zeile mit der 17. Spalte als rechte untere Ecke haben, so braucht man nur zu schreiben:
>100 CALL RAHMEN(2,2,15,17)

TIPS & TRICKS

```
30037 SUB SICHERN
30038 ON BREAK NEXT :: CALL
CLEAR :: TRY=1 :: CALL SCREE
N(2)
30039 NRA=11111111 :: ! KANN
 BELIEBIGE NR. SEIN.
30040 FOR F=0 TO 14 :: CALL
COLOR(F,11,2,3,7,16,4,7,16):
: NEXT F
30041 DISPLAY AT(5,5):"GEBEN
 SIE BITTE IHRE         CODEN
UMMER AN !!"
30042 DISPLAY AT(10,12):"___
_____" :: ACCEPT AT(9,12)VAL
IDATE("1234567890")SIZE(8):N
R
30043 TRY=TRY+1 :: IF NR=NRA
 THEN 30049 :: DISPLAY AT(18
,4):"FALSCHE EINGABE !!"
30044 IF TRY<=2 THEN 30042 :
: IF TRY>3 THEN 30045 :: DIS
PLAY AT(20,4):"SIE HABEN NOC
H EINEN VERSUCH" :: GOTO 300
42
30045 CALL CLEAR :: FOR F=5
TO 9 :: CALL COLOR(F,7,2)::
NEXT F
30046 PRINT TAB(3);"SIE SIND
 NICHT BERECHTIGT" :: PRINT
TAB(3);"********************
*****" :: PRINT
30047 PRINT TAB(3);"DAS GERA
ET WIRD GESPERRT!" :: PRINT
: : : : : : : :
30048 CALL SOUND(-50,444,2):
: GOTO 30048
30049 CALL CLEAR :: DISPLAY
AT(12,5):" B E R E C H T I G
 T " :: FOR T=0 TO 30 :: CAL
L SOUND(-50,1110,T):: NEXT T
30050 CALL CLEAR :: SUBEND
```

Dieses Unterprogramm soll den Zugriff auf bestimmte Teile eines Programms erschweren, um Daten, die nicht für jeden bestimmt sind, zu schützen. Will man an die Daten heran, so muß man eine Schlüsselzahl eingeben, die zwischen 0 und 99999999 liegen kann. Im Programm ist sie in der dritten Zeile enthalten. Sie ist als NRA definiert.
Insgesamt hat man 3 Versuche, die richtige Zahl einzugeben, um auch Fehleingaben berücksichtigen zu können. Beim 3. und letzten Mal wird der Programmablauf gesperrt, der Computer befindet sich in einer unendlichen Schleife. Das Programm läßt sich nicht mit FCTN CLEAR unterbrechen. Schuld daran ist das ON BREAK NEXT in Zeile 30038, das diese Tastenfunktion unbrauchbar macht.
Man kann nur noch, um das Programm zu stoppen, QUIT drücken. Natürlich hätte man auch die QUIT Funktion durch CALL LOAD (–31806,16) außer Funktion setzen können, jedoch müßte man dann entweder die Konsole abschalten oder das Modul rausziehen, wodurch aber die Hardware unnötig beansprucht wird, deshalb wurde hier darauf verzichtet.
Für Besitzer einer Diskettenstation bietet sich noch eine besondere Anwendungsmöglichkeit an. Entfernt man nämlich das SUB SICHERN und das SUBEND aus dem Programm, speichert es auf Diskette mit besonderem Inhalt, kann man hier abfragen, ob der Abfragende berechtigt dazu ist oder nicht, so daß er je nachdem dann ein Menü aufrufen kann, welches diese Dateien zugängig macht.
Noch ein Tip: Damit die Schlüsselzahl nicht sofort beim Listen gefunden wird, könnte man sie in viele Variable zerlegen, die dann zu einer Funktion zusammengefaßt die Schlüsselzahl ergeben, z.B. NRA=A+2B/R-DE+P 2.

TIPS & TRICKS

80 KB IN DER KONSOLE FÜR GANZE 200 MARK

Die erste von mir angestrebte Hardware-Entwicklung sollte eine Speichererweiterung sein und folgende Voraussetzungen erfüllen:

1. Wenig kosten.
2. In die Konsole integrierbar sein.
3. Einen niedrigen Stromverbrauch haben.
4. Die Platine sollte übersichtlich bleiben. (Es sollen schon Leute über dem Zusammenbau von RAM-Platinen mit der Lupe und dem Lötkolben in der Hand eingeschlafen sein)

Alle diese Voraussetzungen konnten nur von dynamischen RAMs erfüllt werden, trotz der zu erwartenden Anpassungsproblemen.
Meine Wahl fiel auf eine 64K-Platine, die derzeit im Handel für 195,– DM angeboten wird, 8x4164, Europa-Format, anpassungsfähig, beschrieben in Elektor Nr136 vom 4/82 (16K dynamische RAM-Karte) und Nr153 vom 9/83 (Erweiterung auf 64K).
Also, zusammenbauen, ansteuern über MEMEN, DBIN und WE und anschließen. Nach Einschalten: ein Fiasko. Der 99er stieß den Fremdkörper erbarmungslos ab und meldete sich in Kyrillisch.
Nach einer schlaflosen Nacht zwischen Oszilloskop und Periodenzähler kam dann die Lösung. Dieses verdammte Ding arbeitet mit einem verdrehten Adressbus! Also Anschlüsse umlöten und einschalten. Extended-Basic anwählen, SIZE eintasten und es kam die langerwartete Meldung 24448 BYTES OF PROGRAM SPACE FREE. Ein RUN-Versuch holte mich dann auf den harten Boden der Realitäten zurück. Der 99er stürzte ins Bodenlose. Nach zwei weiteren durchtasteten Nächten mit EASY-BUG merkte ich, daß nur jedes ungerade Byte geschrieben wurde. Dies war noch eine Eigenart des 99er, der sich als 16-Biter den Luxus erlaubt, gleich zwei Bytes hintereinander zu lesen, und zwar ohne Pause.
Die dynamischen RAMs hatten also gar keine Chance, sich nach dem ersten Byte zu erholen, um sich die Adresse des zweiten abzuholen!
Glücklicherweise gibt es nun auf dieser Platine zwei freie 3-fach AND-Gatter sowie zwei Flip-Flops. Also zwei Flip-Flops und ein Gatter zusammen mit MEMEN (reset), A15 (erkennen zweiter Lese-Zyklus) und 03 (Clock-Takt 3) ergeben eine Erholungspause von 300ns (precharge) für die RAMs.
Nun war der Erfolg total. Der 99er lief, als sei er mit dieser Platine geboren. Betrieb absolut fehlerfrei. Aber wer wird so einfach 32K verschenken, wenn er deren 64 zur Verfügung hat? Banking heißt das Stichwort. Da gibt es ja noch ein freies Gatter auf der Platine und das muß man ausnützen. Also zuerst LOWE-Memory von)2000 auf)0000 legen (bei Zugriff auf)2000-Platine , A2 abschalten)
Dann A0 zum Umschalten benutzen:
A0=0–––LOW-Memory1 +HIGH-Memory1
A0=1–––LOW-Memory2 +HIGH-Memory2
Dies kann mit einem einfachen Schalter bewerkstelligt werden. Da wir aber nicht alle als Weichensteller geboren werden, geht es wahrscheinlich besser mit einigen ICs (für die restlichen 5,– DM) und über DC1-DC4.
Damit kann man dann die vier LOW+HIGH-Memorys auch noch getrennt schalten. Jedenfalls wette ich einen Kasten Bier gegen denjenigen, der ein besseres Preis-Leistungsverhältnis zu bieten hat.
Hier noch ein kleiner Hardware-Trick zum schonenden Umgang mit dem USER-PORT:

BILLIGER GEHT'S NIMMER – WETTEN?

Man nehme einen 50poligen Flachstecker mit geraden Pins

- Säge ihn ab auf 44 Pole wie gehabt
- Stecke ihn auf den USER-PORT
- Löte seine Hardware an eine 64polige DIN 41617 Buchsenleiste und stecke diese nun auf die Pins des Flachsteckers

(Dies erspart auch die Kneifzange beim Hardware-Testen)
Übrigens, die nächste Hardware-Entwicklung ist schon geplant. Falls alles klappt und mein 99er kommenden Winter überlebt, wird er sich wahrscheinlich im Frühjahr eines Disk-Kontrolers erfreuen können. Dieser soll wiederum, getreu dem Motto billiger geht's nimmer, so um die 100,– DM kosten.
Zum Schluß möchte ich noch mit einem alten Aberglauben aufräumen. Nämlich dem, daß man in der LOW-Memory keine Basic-Programme ablegen kann. Der Weg ist folgender:

```
)NEW
)CALL INIT
)100 REM
)SAVE MINIMEM
```

Von jetzt an vergißt die LOW-Memory ihre Assembler-Affinitäten und schluckt brav:

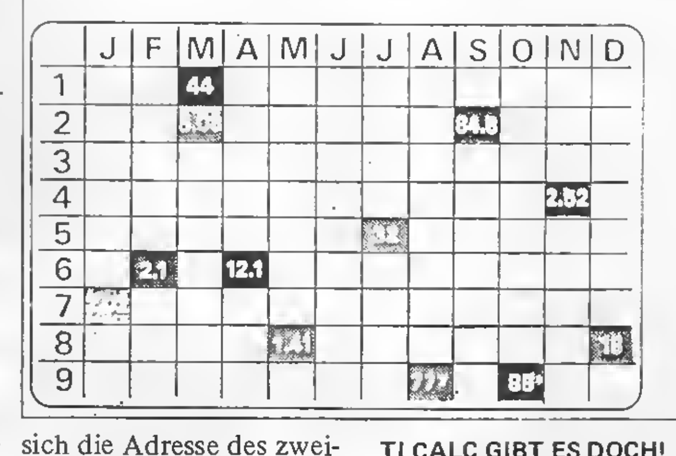
```
)SAVE EXPMEM1
)OLD EXPMEM1
```

Armand Marx

ORDINATEUR FAMILIAL TEXAS INSTRUMENTS

TI-CALC

ELEKTRONISCHE KALKULATIONSTABELLE FÜR DEN TEXAS INSTRUMENTS HOME COMPUTER 99/4A



TI CALC GIBT ES DOCH!

In Frankreich wurde unser Leser Marx fündig! Das bereits angekündigte, dann aber angeblich verschrottete TI-Calc gibt es. In Deutsch!
Kostet keine 150 Mark und läuft auf jedweder Konfiguration! Programmierer: Don Douglas Bond Ass. Ltd., 2, Champden Terrace, Linden Gardens, Chiswik, London W 4, England. Offensichtlich produziert in Italien, mit deutschem Handbuch.
Also: Bei einem Frankreichbesuch: Augen auf!



TIPS & TRICKS

PINBELEGUNG AUF EINEN BLICK

Viele TI-User sind – der Not gehorchend – zu Lötkolben-Rastellis geworden. Peripherie, die nicht mehr erhältlich ist, wird selbst gebaut. Immer wieder erreicht uns daher die Frage nach der Pin-Belegung des I/O-Port. Hier ist sie. Die Leitungen A0 bis A15 sind Adressleitungen, D 0 bis D 7 Datenleitungen. Und wenn wir schon dabei sind, hier auch noch gleich anschließend die Belegung des internen Steckers in der Modulbox. Doch Vorsicht: Die I/O-Pins führen direktemang ins Herz der Maschine, Fehler wirken sich entsprechend übel aus. Der Schrottplatz winkt!

PIN-Nr.	Mnemonic	Erläuterung
01		+ 5 Volt geregelt
02		+ 5 Volt geregelt
03	GND	Masse (Logikmasse)
04	Ready A	System bereit
05	GND	wie PIN 03
06	Reset	Systemgesteuerter Reset (Aktiv=LOW)
07	GND	wie PIN 03
08	SCLK	System-Takt
09	LCP	CPU-Indikator (1=TI 99/4; 0=TI 99/4A)
10	Audio	Audiosignal
11	RBDENA	Remote data bus driver (Aktiv=LOW)
12	PCBEN	Platinenansteuerung für „burn-in" (Aktiv=HIGH)
13	HOLD	CPU HOLD Anforderung (nur TI 99/4A)
14	IAQHA	Logisch ODER bei IAQ HOLDA (nur TI 99/4A
15	SENILA	Interrupt Level A Sense Enable (für PCC vorgesehen)
16	SENILB	Interrupt Level B Sense Enable (für PCC vorgesehen)
17	INTA	Interrupt Level A (Unterbrechungsebene; Aktiv= LOW)
18	LOAD	LOW bei TI 99/4 Speichererweiterung HIGH bei freiem Speicherplatz TMS9900 LOAD input (nur TI 99/4A)
19	D7	System Data Bus, least significant bit
20	GND	wie PIN 3
21	D5	System Data Bus
22	D6	System Data Bus
23	D3	System Data Bus
24	D4	System Data Bus
25	D1	System Data Bus
26	D2	System Data Bus
27	GND	wie PIN 3
28	DO	System Data Bus, most significant Bit
29	A14.A	Adress Bit
30	A15/ CRUOUT.A	Adress Bit, least significant Bit
31	A12.A	Adress Bit
32	A13.A	Adress Bit
33	A10.A	Adress Bit
34	A11.A	Adress Bit
35	A08.A	Adress Bit
36	A09.A	Adress Bit
37	A06.A	Adress Bit
38	A07.A	Adress Bit
39	A04.A	Adress Bit
40	A05.A	Adress Bit
41	A02.A	Adress Bit
42	A03.A	Adress Bit
43	A00.A	Adress Bit
44	A01.A	Adress Bit
45	AMB.A	Adress Bit, normalerweise "HIGH"
46	AMA.A	Adress Bit, normalerweise "HIGH"
47	GND	wie PIN 03
48	AMC.A	Adress Bit, most significant Bit, normalerweise "HIGH"
49	GND	wie PIN 03
50	CLKOUT	CPU Takt, Aktiv=LOW
51	CRUCLK.A	CPU-Taktausgang, Aktiv+LOW
52	DBIN.A	Data Bus Dir'tn, HIGH=CPU–Read
53	GND	wie PIN 03
54	WE.A	CPU Schreibsteuerung (LOW)
55	CRUIN	CRU Input Data (HIGH)
56	MEMEN.A	Speicheranfrage (Aktiv+LOW)
57		–12 Volt geregelt
58		–12 Volt geregelt
59		+12 Volt geregelt
60		+12 Volt geregelt

I/O PORT PIN ASSIGNMENT
PERIPHERAL I/O PORT

PIN	DESCRIPTION	PIN	DESCRIPTION
1	+5 VOLT	2	SBE (Speech Select)
3	RESET	4	EXT INT
5	A5	6	A10
7	A4	8	A11
9	DBIN	10	A3
11	A12	12	READY/HOLD
13	LOAD	14	A8
15	A13	16	A14
17	A7	18	A9
19	A15	20	A2
21	GND	22	CRU CLK
23	GND	24	0 3
25	GND	26	WE
27	GND	28	MBE
29	A6	30	A1
31	A0	32	MEMEN
33	CRU IN	34	D7
35	D4	36	D6
37	D0	38	D5
39	D2	40	D1
41	HOLD/IAQ	42	D3
43	–5 VOLT	44	SPEECH

TIPS & TRICKS

DER DEBUGGER UND SEINE MÖGLICHKEITEN

Aufgrund der Tatsache, daß die Zahl der Editor/Assembler-Besitzer laufend steigt und somit ebenfalls (hoffentlich) der Assembler Programmierer, habe ich mich entschlossen, Teile des Debuggers näher zu erläutern.
Nach genauem Studium, sowohl der englischen als auch der deutschen Beschreibung, zeigten sich einige Mängel, die ich hiermit beheben möchte. Der erste Befehl auf den ich näher eingehen möchte, gestattet das Auslesen des Assemblermoduls, sowie das Inspizieren sämtlicher Grom-Module.
Mit dem Kommando M G6000,7800 zeigt sich der Inhalt des E/A-Moduls, das nach dem Laden des Debuggers auch aus dem Gromport entfernt werden kann. Jetzt könnte z.B. das Statistik-Modul ins Port gesteckt und ausgelesen werden, wenn nicht beim Einschub eines Moduls ein Resetimpuls auftreten würde.
Dieser Reset kann auf einfache Weise bei Modulen, die nur mit Groms (Graphic Read Only Memory) bestückt sind, außer Kraft gesetzt werden. Die nachfolgende Liste führt einige dieser Module auf:

Ti-Writer
Text- und Dateiverwaltung
Editor/Assembler
Datenverwaltung und Analyse
Statistic
Multiplan
Buchungsjournal
Wumpus
Speech-Editor
Amazing
Adventure usw.

Sollten Sie im Zweifel darüber sein, ob Ihr Modul ein reines Grom-Modul ist, genügt das Öffnen des Gehäuses.
Treten dabei nur ICs zutage mit maximal 16 Anschlüssen, so besteht kein Zweifel, daß es sich um ein nur mit Groms bestücktes Modul handelt. Zum Auslesen des Moduls genügt das Überkleben des Pins 1 mit einem Stück Tesafilm. Dieser Pin befindet sich, wenn Sie Ihr Modul von vorne betrachten (Steckerseite, die in den Gromschlitz gesteckt wird), rechts unten. Nach dieser Behandlung können Sie Ihr Modul jetzt in das Grom-Port stecken und auslesen.

MODIFIKATION OHNE EINGRIFFE

Diese Modifikation kann ohne irgendwelche Eingriffe ins Modul auch von außen durchgeführt werden.
Die nachfolgende Tabelle zeigt die gültigen Adreßbereiche der Groms:

M G6000,7800	Grom 1
M G8000,9800	Grom 2
M GA000,B800	Grom 3
M GC000,D800	Grom 4
M GE000,F800	Grom 5

Noch ein Tip zum Auslesen.
Leider hat sich gezeigt, daß manchmal trotz „Spezialbehandlung" ein Reset auftritt. Dieser Fehler kann durch folgende Punkte weitgehend eingedämmt werden.

1. Start des Debuggers
2. Befehl M 0,FFFF
3. Entfernen des E/A-Moduls
4. Einstecken des auszulesenden Moduls
5. Anhalten der Auflistung mit FCTN X
6. Auslesen laut obiger Tabelle
7. Bei Absturz Neu beginn mit Punkt 1

DSR AUSLESEN

Der zweite Befehl, auf den ich näher eingehen möchte, bezieht sich auf das Auslesen der Peripherie-Roms (DSR).
Die meisten Karten, u.a. P-Code, RS232, Controller, beinhalten ein Rom mit ihrem Betriebssystem. Diese Karten liegen auf folgenden CRU-Adressen.

1100	Controller
1300	RS232 I
1500	RS232 II
1800	Thermalprinter
1F00	P-Code

Der Adreßbereich des DSR-Roms befindet sich bei 4000 – 5FFF. Je nach Notwendigkeit wird vom System eines der Peripheriegeräte zu- oder abgeschaltet. Dieses Zuschalten bei Gebrauch bietet gegenüber anderen Computersystemen den Vorteil, daß bei einer Peripherieerweiterung kein Rambereich verloren geht.
Wie kann man nun z.B. das Betriebssystem der RS232 I auslesen?
Mit dem Kommando C 1300,1 <Enter> erfolgt die Ausgabe C 1300,XXXX die mit einer Eingabe von 1 <Enter> abgeschlossen wird. Nun können Sie das Rom im Bereich 4000–5FFF auslesen.

AUCH FÜR MINIMEMORY

Der Befehl C 1300,1 bewirkt dabei, daß nur ein Bit ab der Cru-Adresse 1300 Hex ausgegeben werden kann. C 1300,0 würde eine 16 Bit breite Ausgabe ab 1300 erlauben.
Bevor Sie nun ein anderes Rom auslesen, muß erst das Cru-Bit bei 1300 zurückgesetzt werden. Das Rücksetzen erfolgt mit:

	C 1300,1<Enter>
Ausgabe	C 1300,XXXX
Eingabe	0 <Enter>



Sollte jemand seine spielerische Ader mit dem Ein- bzw. Ausschalten der Peripherie-Leuchtdioden befriedigen wollen, so kann er dies durch das Senden eines Bits auf folgenden Adressen:

1100	/1100	Controller
130E	/1300	RS232 I
150E	/1500	RS232 II
1F00	/1F00	P-Code
Led ein bzw. aus	/ Bit setzen zum Auslesen der DSR (DSR=Device Service Routine).	

Aus dieser Tabelle wird erkennbar, daß sowohl bei der Controller- als auch der P-Code-Karte nur ein Bit für das Aufleuchten der Led und das gleichzeitige Auslesen gesetzt werden muß.
Zum Schluß möchte ich noch anmerken, daß auch Minimemory-Besitzer den Debugger starten und somit verwenden können. Eine Kopie von einem Freund oder Bekannten wird sich bestimmt als nützlich erweisen.

Reiner Ziegler

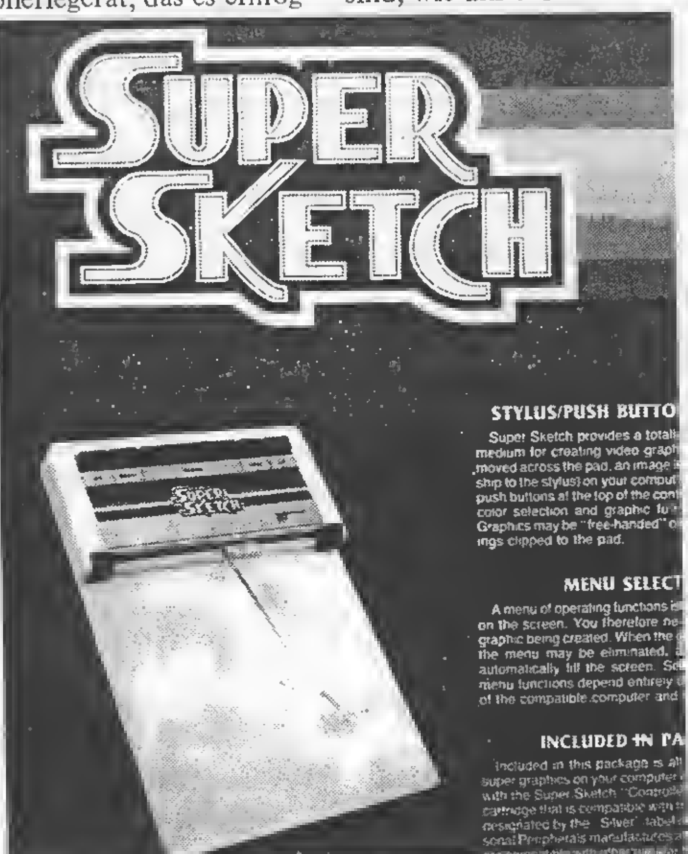
GRAPHIKTABLETT

Nachdem Graphiktabletts für viele Heimcomputer schon seit einiger Zeit im Handel erhältlich sind freuten wir uns sehr, als uns die Firma Hega-Soft in München nach langer Zeit endlich das Graphiktablett für den TI 99/4A für Testzwecke zur Verfügung stellen konnte.
Der Lieferumfang besteht aus dem Graphiktablett selbst, dem Anschlußmodul sowie einem kleinen englischen Handbuch. Weiteres Zubehör ist nicht notwendig. Dieses Tableau läuft bereits mit der Grundversion der Konsole. Der Anschluß ist kinderleicht. Das Anschlußmodul in den Modulschacht gesteckt und einen Zeichenstift in den Zeichenarm eingeklemmt. Am Bildschirmmenü wählen wir nun die Option 2 für das Graphiktablett aus. Alle weiteren Operationen werden dann mit dem Kontrollarm des Graphiktabletts sowie den Tasten am Tablett selbst angewählt.

Dieses Graphiktablett, genannt Supersketch, spricht sowohl Kinder als auch Erwachsene an. Es stellt eine der einfachsten Möglichkeiten, sich mit dem Gebrauch von Computergraphik vertraut zu machen, dar. Sämtliche Kommandos werden über eine Menüleiste am Bildschirmrand angewählt, eine der einfachsten Möglichkeiten, sich im Umgang mit Computern zu üben. Sie können jedoch nicht nur einfache Graphikbilder mit dem Supersketch erzeugen, sondern auch professionelle Balken- und Tortengraphik.
Im begefügten Starterkit sind sechs verschiedene einfache Graphikbilder vorgegeben, um die ersten Malversuche zu erleichtern.

Im Gegensatz zu den Supersketchversionen, die auch für andere Computer angeboten werden, können Sie mit der TI-Version die Zeichenbilder auch abspeichern. Bisher geht das leider nur auf Kassette. Sie wählen zum Speichern aus dem Expertmenü die Save- bzw. zum Einladen die Load-Routine an. Alles weitere funktioniert genauso, als ob sie die Kassettenroutine aus einem Basicprogramm heraus angewählt hätten; also wie gewohnt.
Die Funktion RW-FF erlaubt es, den Kassetten-Vor- bzw. Rücklauf anzusteuern.

Gesamtbewertung: Supersketch ist ein neues Peripheriegerät, das es ermöglicht, ohne Computerkenntnisse bereits ansprechende Bilder auf dem Bildschirm zu erzeugen. Besonders hervorzuheben ist, daß dieses Gerät bereits mit der Grundversion der Konsole funktioniert. Der Bedienungsarm schien uns am Anfang etwas wacklig; doch dieses Problem ist bei der Texas Instrument-Version elegant gelöst worden. Um gerade Linien zu ziehen, stehen die Zusatzbefehle des Expertmenüs zur Verfügung. Somit kann man dem Arm keinen Minuspunkt mehr geben. Positiv anzumerken ist, daß die erzeugten Bilder abgespeichert werden können, jedoch leider, und das ist negativ, nur auf Kassette und nicht auf Diskette. Unserer Meinung nach ein interessantes Zusatzgerät, dessen Kauf sich durchaus lohnen kann.



PROGRAMME MIT HARDCOPY

Zu den Programmen 3-D-World und Graphikmaster sind, wie uns der deutsche Lizenznehmer mitteilte, nunmehr auch für die gängigsten Druckermodelle Hardkopie-Routinen in Maschinensprache erhältlich. Der empfohlene Verkaufspreis der Graphikprogramme incl. Hardkopie-Routine soll bei ca. 130 DM liegen. Bezugsquellen: Der einschlägige Fachhandel.

ZEITMESSUNG MIT DEM TIX

Auch denjenigen Besitzern eines TI 99/4A, die nur über das Extended Basic Modul verfügen, ist es möglich, über ein kleines Programm eine relativ genaue Zeitmessung vorzunehmen. Es wurde als Unterprogramm geschrieben und läßt sich leicht in vorhandene Programme einbauen. Der Aufruf erfolgt durch den Befehl CALL UHR. Man nutzt dabei die besonderen Eigenschaften des Speicherplatzes 33657 der CPU, dem VDP-INTERRUPTTIMER. Um in eine höhere Adresse als 32767 zu PEEKen, muß hiervon zunächst 65536 abgezogen werden; daher der Befehl CALL PEEK (−31879, ZN).
Der Inhalt dieses Speicherplatzes wird 50(!)mal in der Sekunde um eins erhöht und durchläuft dabei die Werte zwischen 0 und 255. Diese Erhöhung erfolgt unabhängig vom laufenden Programm. Die Anzeige erfolgt in Stunden (=H), Minuten (=M) und Sekunden (=S). Andere Anzeigemöglichkeiten erreicht man durch Änderung der Zeile 10030. Um Ungenauigkeiten durch Zeitsprünge zu vermeiden, ist lediglich sicherzustellen, daß die Abfrage des Speicherinhaltes spätestens alle 5 Sekunden erfolgt. Dies läßt sich jedoch durch geeignete Platzierung des Unterprogrammaufrufes auf einfache Weise realisieren.

```
10!TI-UHR
20!von Berthold Wölfel-
   schneider
   Im Hopfengarten 17
   6200 Wiesbaden
10000 SUB UHR
10010 CALL PEEK (
(−31879,ZN)::Z=Z+ZN
−ZA−255*(ZA>ZN)::
IF Z<3000 THEN 10030
10020 Z=Z−3000::M=M
+1::IF M=60 THEN H=H
+1::M=0
10030 S=INT(Z/50)::ZA
=ZN::DISPLAY AT
(24,12):H;":";M;":"S
::SUBEND
```

Berthold Wölfelschneider

WAHL-AUSWERTUNG

```
100 REM  *******************
*
110 REM  *
*
120 REM  *   COPYRIGHT BY
*
130 REM  *   NORMAN & LARS
*
140 REM  *      KELLER
*
150 REM  *  BEARBEITET VOM
*
160 REM  *     TEAM DES
*
170 REM  * AKTUELL VERLAGES
*
180 REM  *     MUENCHEN
*
190 REM  *BENOETIGTE GERAETE
*
200 REM  * TI 99/4 A KONSOLE
*
210 REM  *     TI BASIC
*
220 REM  * DISK.OD.CASSETTE
*
230 REM  *
*
240 REM  * SPEICHERBELEGUNG
*
250 REM  *   6498  BYTES
*
260 REM  *******************
*
270 CALL CLEAR
280 DIM S(5),R(5)
290 F$="FFFFFFFFFFFFFFFF"
300 CALL CHAR(64,"3C4299A1A1
99423C")
310 CALL CHAR(91,"0024003C42
427E42")
320 CALL CHAR(93,"0024003C42
42423C")
330 CALL CHAR(123,"003824282
424282")
340 CALL CHAR(124,"004200424
242423C")
350 CALL SCREEN(15)
360 PRINT ".....************
****"
370 PRINT ".....*...........
...*"
380 PRINT ".....*WAHLAUSWERT
UNG*"
390 PRINT ".....*...........
...*"
400 PRINT ".....************
****"
410 PRINT :"......$BY.NORMAN
.AND"
420 PRINT ".........LARS KEL
LER"::::
430 PRINT "...BITTE TASTE DR
öCKEN"::::
440 CALL KEY(0,K,ST)
450 IF ST=0 THEN 440
460 GOSUB 2490
470 CALL CLEAR
480 PRINT
490 INPUT "1.PARTEI:":Z1$
500 IF LEN(Z1$)>5 THEN 490
510 PRINT
520 INPUT "2.PARTEI:":Z2$
530 IF LEN(Z2$)>5 THEN 520
540 PRINT
550 INPUT "3.PARTEI:":Z3$
560 IF LEN(Z3$)>5 THEN 550
570 PRINT
580 INPUT "4.PARTEI:":Z4$
590 IF LEN(Z4$)>5 THEN 580
600 PRINT
610 INPUT "5.PARTEI:":Z5$
620 IF LEN(Z5$)>5 THEN 610
630 PRINT
640 INPUT "STIMMEN:":K
650 IF K<>INT(K)THEN 640
660 INPUT "UNGöLTIG:":N
670 IF N<>INT(N)THEN 660
680 IF K=N THEN 2460
690 PRINT
700 INPUT "STIMMEN FöR 1.PAR
TEI:":O
710 IF O<>INT(O)THEN 700
720 PRINT
730 INPUT "STIMMEN FöR 2.PAR
TEI:":P
740 IF P<>INT(P)THEN 730
750 PRINT
760 INPUT "STIMMEN FöR 3.PAR
TEI:":Q
770 IF Q<>INT(Q)THEN 760
780 PRINT
790 INPUT "STIMMEN FöR 4.PAR
TEI:":Z
800 IF Z<>INT(Z)THEN 790
810 PRINT
820 INPUT "STIMMEN FöR 5.PAR
TEI:":X
830 IF X<>INT(X)THEN 820
840 M=K-N
850 IF M<>O+P+Q+Z+X THEN 243
0
```

```
860 R1=O/M
870 S(1)=INT(1000*R1)/10
880 R2=P/M
890 S(2)=INT(1000*R2)/10
900 R3=Q/M
910 S(3)=INT(1000*R3)/10
920 R4=Z/M
930 S(4)=INT(1000*R4)/10
940 R5=X/M
950 S(5)=INT(1000*R5)/10
960 IF S(1)+S(2)+S(3)+S(4)+S
(5)<>100 THEN 2660
970 P1$=STR$(S(1))&CHR$(37)
980 P2$=STR$(S(2))&CHR$(37)
990 P3$=STR$(S(3))&CHR$(37)
1000 P4$=STR$(S(4))&CHR$(37)
1010 P5$=STR$(S(5))&CHR$(37)
1020 CALL CLEAR
1030 CALL SCREEN(12)
1040 A$="STIMMENVERTEILUNG"
1050 B$="================="
1060 FOR Z=1 TO LEN(A$)
1070 CALL HCHAR(1,Z+8,ASC(SE
G$(A$,Z,1)))
1080 NEXT Z
1090 FOR Z=1 TO LEN(B$)
1100 CALL HCHAR(2,Z+8,ASC(SE
G$(B$,Z,1)))
1110 NEXT Z
1120 T1=INT(20*R1)
1130 U1=20*R1
1140 T2=INT(20*R2)
1150 U2=20*R2
1160 T3=INT(20*R3)
1170 U3=20*R3
1180 T4=INT(20*R4)
1190 U4=20*R4
1200 T5=INT(20*R5)
1210 U5=20*R5
1220 IF T1<>U1 THEN 1230 ELS
E 1240
1230 GOSUB 1910
1240 CALL CHAR(94,F$)
1250 CALL COLOR(8,2,12)
1260 FOR A=23 TO 24-T1 STEP
-1
1270 CALL HCHAR(A,2,94,5)
1280 NEXT A
1290 IF T2<>U2 THEN 1300 ELS
E 1310
1300 GOSUB 1960
1310 CALL CHAR(144,F$)
1320 CALL COLOR(15,7,12)
1330 FOR B=23 TO 24-T2 STEP
-1
1340 CALL HCHAR(B,8,144,5)
1350 NEXT B
1360 IF T3<>U3 THEN 1370 ELS
E 1380
1370 GOSUB 2010
1380 CALL CHAR(128,F$)
1390 CALL COLOR(13,8,12)
1400 FOR C=23 TO 24-T3 STEP
-1
1410 CALL HCHAR(C,14,128,5)
1420 NEXT C
1430 IF T4<>U4 THEN 1440 ELS
E 1450
1440 GOSUB 2060
1450 CALL CHAR(136,F$)
1460 CALL COLOR(14,13,12)
1470 FOR D=23 TO 24-T4 STEP
-1
1480 CALL HCHAR(D,20,136,5)
1490 NEXT D
1500 IF T5<>U5 THEN 1510 ELS
E 1520
1510 GOSUB 2110
1520 CALL CHAR(96,F$)
1530 CALL COLOR(9,14,12)
1540 FOR E=23 TO 24-T5 STEP -1
1550 CALL HCHAR(E,26,96,5)
1560 NEXT E
1570 FOR X=1 TO LEN(Z1$)
1580 CALL HCHAR(22-T1,X+1,AS
C(SEG$(Z1$,X,1)))
1590 NEXT X
1600 FOR X=1 TO LEN(Z2$)
1610 CALL HCHAR(22-T2,X+7,AS
C(SEG$(Z2$,X,1)))
1620 NEXT X
1630 FOR X=1 TO LEN(Z3$)
1640 CALL HCHAR(22-T3,X+13,A
SC(SEG$(Z3$,X,1)))
1650 NEXT X
1660 FOR X=1 TO LEN(Z4$)
1670 CALL HCHAR(22-T4,X+19,A
SC(SEG$(Z4$,X,1)))
1680 NEXT X
1690 FOR X=1 TO LEN(Z5$)
1700 CALL HCHAR(22-T5,X+25,A
SC(SEG$(Z5$,X,1)))
1710 NEXT X
1720 FOR Y=1 TO LEN(P1$)
1730 CALL HCHAR(24,Y+1,ASC(S
EG$(P1$,Y,1)))
1740 NEXT Y
1750 FOR Y=1 TO LEN(P2$)
1760 CALL HCHAR(24,Y+7,ASC(S
EG$(P2$,Y,1)))
```

```
1770 NEXT Y
1780 FOR Y=1 TO LEN(P3$)
1790 CALL HCHAR(24,Y+13,ASC(
SEG$(P3$,Y,1)))
1800 NEXT Y
1810 FOR Y=1 TO LEN(P4$)
1820 CALL HCHAR(24,Y+19,ASC(
SEG$(P4$,Y,1)))
1830 NEXT Y
1840 FOR Y=1 TO LEN(P5$)
1850 CALL HCHAR(24,Y+25,ASC(
SEG$(P5$,Y,1)))
1860 NEXT Y
1870 CALL KEY(0,K,ST)
1880 IF ST=0 THEN 1870
1890 IF (K=74)+(K=106)=0 THE
N 2780
1900 GOTO 470
1910 QX=(U1-T1)*8
1920 CD=95
1930 SP=23-T1
1940 LD=2
1950 GOTO 2150
1960 QX=(U2-T2)*8
1970 CD=145
1980 SP=23-T2
1990 LD=8
2000 GOTO 2150
2010 QX=(U3-T3)*8
2020 CD=129
2030 SP=23-T3
2040 LD=14
2050 GOTO 2150
2060 QX=(U4-T4)*8
2070 CD=137
2080 SP=23-T4
2090 LD=20
2100 GOTO 2150
2110 QX=(U5-T5)*8
2120 CD=97
2130 SP=23-T5
2140 LD=26
2150 IF QX=0 THEN 2240
2160 IF QX<=1 THEN 2260
2170 IF QX<=2 THEN 2280
2180 IF QX<=3 THEN 2300
2190 IF QX<=4 THEN 2320
2200 IF QX<=5 THEN 2340
2210 IF QX<=6 THEN 2360
2220 IF QX<=7 THEN 2380
2230 IF QX<8 THEN 2400
2240 CALL CHAR(CD,"")
2250 GOTO 2410
2260 CALL CHAR(CD,"000000000
00000FF")
2270 GOTO 2410
2280 CALL CHAR(CD,"000000000
000FFFF")
2290 GOTO 2410
2300 CALL CHAR(CD,"000000000
0FFFFFF")
2310 GOTO 2410
2320 CALL CHAR(CD,"00000000F
FFFFFFF")
2330 GOTO 2410
2340 CALL CHAR(CD,"000000FFF
FFFFFFF")
2350 GOTO 2410
2360 CALL CHAR(CD,"0000FFFFF
FFFFFFF")
2370 GOTO 2410
2380 CALL CHAR(CD,"00FFFFFFF
FFFFFFF")
2390 GOTO 2410
2400 CALL CHAR(CD,"FFFFFFFFF
FFFFFFF")
2410 CALL HCHAR(SP,LD,CD,5)
2420 RETURN
2430 CALL SOUND(100,440,2)
2440 PRINT "EINGABE öBERPRöF
EN:PARTEI-..STIMMENZAHL UNGL
EICH ZAHL...DER GöLTIGEN STI
MMEN !"
2450 GOTO 640
2460 CALL SOUND(100,440,2)
2470 PRINT "IHRE EINGABE IST
 FALSCH: DIEGESAMTSTIMMENZAH
L DARF NICHTNULL SEIN!"
2480 GOTO 640
2490 REM  PROGRAMMBENUTZUNG
2500 CALL CLEAR
2510 PRINT "DIESES PROGRAMM
STELLT DURCH EIN SCHAUBILD D
IE STIMMEN- VERTEILUNG DAR,D
IE AUS EINER WAHL RESULTIERT
."
2520 PRINT "ES KüNNEN 5 VERS
CHIEDENE    'PARTEIEN' ANGEG
EBEN WERDEN.AUßERDEM WERDEN
DIE JEWEI-  LIGEN PROZENTUAL
EN STIMMEN- ANTEILE AUSGEDRU
CKT."
2530 PRINT "DIE LäNGE DER 'P
ARTEI'-NAMEN DARF NICHT MEHR
 ALS 5 BUCH-STABEN BETRAGEN.
"
2540 PRINT "BEI FALSCHEN ZAH
LENEINGABEN WERDEN ENTSPRECH
ENDE HIN-   WEISE AUSGEGEBEN
."
```

```
2550 PRINT "NACHDEM DAS SCHA
UBILD ER-   SCHIENEN IST, KÜ
NNEN SIE 'J'FÖR NOCHMALIGE P
ROGRAMMAN-  WENDUNG DRÖCKEN.
 ANSONSTEN  WIRD DAS PROGRAM
M BEENDET."
2560 PRINT "BITTE TASTE DRÖC
KEN !"
2570 CALL KEY(0,K,ST)
2580 IF ST=0 THEN 2570
2590 CALL CLEAR
2600 PRINT "WICHTIG !!!"
2610 PRINT "ES IST ZU EMPFEH
LEN, DIE     'PARTEI'-NAMEN I
N GROäBUCH- STABEN EINZUGEBE
N, DA SONST DAS SCHAUBILD ZU
 BUNT WIRD."
2620 PRINT "ZUM PROGRAMMSTAR
T...........TASTE DRÖCKEN !"
2630 CALL KEY(0,K,ST)
2640 IF ST=0 THEN 2630
2650 RETURN
2660 R(1)=R1
2670 R(2)=R2
2680 R(3)=R3
2690 R(4)=R4
2700 R(5)=R5
2710 FOR I=1 TO 5
2720 DF(I)=R(I)*10000-(10*IN
T(R(I)*1000))
2730 IF INT(DF(I))>4 THEN 27
40 ELSE 2760
2740 S(I)=S(I)+.1
2750 IF S(1)+S(2)+S(3)+S(4)+
S(5)=100 THEN 2770
2760 NEXT I
2770 GOTO 970
2780 END
```

WAHL

Wenn auch im Aufbau einfach, so ist dies, gerade wegen seiner Schlichtheit, ein sehr schönes BASIC-Programm!!!
Gut geeignet bei Vereinswahlen, es weist verblüffende Effekte vor.

PROGRAMMBESCHREIBUNG ZUM PROGRAMM „WAHLAUSWERTUNG“

Dieses Programm berechnet die prozentuale Stimmenverteilung einer Wahl und stellt sie grafisch – als Balkendiagramm – dar. Es können maximal 5 verschiedene Parteien eingegeben werden, deren Namen über den zugehörigen Balken angezeigt werden.
Aufgrund Platzmangels kann die Prozentangabe nur bis auf eine Stelle nach dem Komma ausgegeben werden.

Zur Anwendung kann das Programm bei Wahlen aller Art kommen, z.B. bei Landtagswahlen, Kommunalwahlen, aber auch bei Wahlen kleineren Umfangs (Sportverein, Club, etc.).

Hier einige direkte Programmerläuterungen:

Zeile 300–340	:	Definition der Umlaute, “ß” und “©”
Zeile 490–830	:	Eingabe-Block
Zeile 840–960	:	Berechnung der prozentualen Stimmanteile
Zeile 1120–1830	:	Aufbau und Anzeige des Balkendiagramms
Zeile 1910–2420	:	Programmroutine zur grafischen Anzeige der Stimmanteile, die sich nicht als „voller“ Block darstellen lassen
Zeile 2660–2760	:	Routine zum Überprüfen, ob Prozentsumme 100 ergibt.

Eine Vielzahl von Einsendungen erreichte uns leider in stark beschädigtem Zustand. Bei einer Reihe von Briefen fehlten die Kassetten oder die Begleitschreiben oder waren diese vertauscht. Dies kam daher, daß die viel zu dünnen Umschläge unterwegs aufgeplatzt waren. Die Deutsche Bundespost hat sich zwar größte Mühe gegeben, die Sendungen noch zu verpacken und zu retten, was noch zu retten war, aber immer ging es halt leider nicht.
Auch waren einige Kassetten und Disketten so stark beschädigt, daß wir sie nicht mehr verwenden konnten. Unsere Bitte: Kassetten und Disketten nach Möglichkeit in Luftpolstertaschen an uns senden, da diese erfahrungsgemäß am wenigsten anfällig sind.

KLETTER-MAX

Ein Monteur muß auf einem Gerüst Brückenteile entfernen und dadurch Punkte sammeln. Er muß sich aber ganz schön beeilen, da das Gerüst nach und nach verschwindet. Desto mehr Brückenteile durch Überlaufen abgebaut wurden, desto schneller demontiert sich das Gerüst selbst. Als zusätzliche Schwierigkeit erscheint von Zeit zu Zeit eine Bombe, begleitet von einem Warngeräusch. Führen Sie nun den Monteur in Richtung der Bombe. Nach dem 2. Signal kann er nun die Bombe durch die Taste ‘2’ entschärfen und kassiert dafür Bonuspunkte. Aber Vorsicht, nach dem 2. Signal kann die Bombe jeden Augenblick explodieren. Ist der Monteur dann in Reichweite der Bombe so ist sein Leben beendet, andernfalls reißt die Bombe „nur“ ein Loch ins Gerüst. Um dem armen Monteur das Leben aber nicht ganz so schwer zu machen, wurden ihm einige Hilfsmittel mit auf den Weg gegeben: Mit der Taste ‘J’ macht der Monteur einen kleinen Sprung über eine Lücke oder in die nächst höhere oder tiefere Ebene. Taste ‘K‘ und Taste ‘L‘ sind ebenfalls für Sprünge reserviert, nur daß die Sprünge jeweils länger ausfallen. Sie haben pro Runde und Taste jeweils 2 Sprünge zur Verfügung. Die Richtung bestimmen Sie mit den Pfeiltasten. Und sollte der Monteur mal abstürzen, ist nicht schlimm, er hat durch Taste ‘1‘ einen Fallschirm zur Verfügung, der ihn sicher in der nächsten Ebene landen läßt, aber nur einmal pro Runde benutzt werden darf. Am Ende des Spieles können Sie sich in eine Bestenliste eintragen und diese abspeichern. Vielleicht erreichen Sie beim nächsten Mal mehr Punkte?

J. Wolters

```
100 REM********************

110 REM*   Kletter-max     *

120 REM* Copyright  by     *

130 REM* Joerg Wolters     *

140 REM*                   *

150 REM* Bearbeitet  vom   *

160 REM*    Team  des      *

170 REM* Aktuell Verlages  *

180 REM*     Muenchen      *

190 REM*                   *

200 REM*Benoetigte Geraete*

210 REM* TI99/4A  Konsole *

220 REM*                   *

230 REM* Speicherbelegung *

240 REM*   10766  Bytes    *
250 REM*                   *

260 REM********************

270 DIM LIST$(20),PUNKTE(20)
280 RANDOMIZE
290 CALL CLEAR
300 PRINT "*****************
**********":"****....KLETTER
-MAX....****":"*************
**************"
310 PRINT "...BY 'JOERG WOLT
ERS":"...516 DUEREN-7"
320 FOR I=1 TO 400
330 NEXT I
340 PRINT "WENN SIE DIE BEST
ENTABELLE":"EINLADEN WOLLEN,
DANN ":"DRUECKEN SIE DIE LEE
RTASTE":"SONST ""S"""
350 CALL KEY(0,A1,B1)
360 IF B1=0 THEN 350
370 IF A1<>32 THEN 430
380 OPEN #1:"CS1",INTERNAL,S
EQUENTIAL,INPUT ,FIXED 192
390 FOR I=1 TO 5
400 INPUT #1:LIST$(I),PUNKTE
(I),LIST$(I+1),PUNKTE(I+1),L
IST$(I+2),PUNKTE(I+2),LIST$(
I+3),PUNKTE(I+3)
410 NEXT I
420 CLOSE #1
430 REM ZEICHEN DEFINIEREN
440 CALL CHAR(128,"0018183C5
A182442")
450 CALL CHAR(129,"7EFF183C5
A182442")
460 CALL CHAR(136,"0081C3FFF
FC381")
470 CALL CHAR(137,"FFFF4224F
F2442FF")
480 CALL CHAR(144,"FFFFFFFF"
)
490 CALL CHAR(153,"FFFFC3C3C
3C3C3C3")
500 CALL CHAR(152,"FFFFC3A59
9A5C3FF")
510 CALL CHAR(43,"0609183C7E
7E3C18")
520 CALL CHAR(42,"95563CFF3C
DE2D4A")
530 RANDOMIZE
540 REM FARBGEBUNG
550 CALL CLEAR
560 CALL SCREEN(2)
570 CALL COLOR(2,10,2)
580 CALL COLOR(13,11,2)
590 CALL COLOR(14,6,2)
600 CALL COLOR(15,13,5)
610 CALL COLOR(16,9,2)
620 PZC=1000
630 CONTR4=1
640 PZ=0
650 RA=1
660 AZ=3
670 REM GERUESTAUFBAU
680 CALL CLEAR
690 ZE=32
700 CONTR5=0
710 READ I
720 READ APA
730 FOR J=1 TO I
740 READ Y,X,W
750 CALL HCHAR(Y,X,137,W)
760 NEXT J
770 READ I
780 FOR J=1 TO I
790 READ Y,X,W
800 CALL HCHAR(Y,X,144,W)
810 NEXT J
820 READ I
830 FOR J=1 TO I
840 READ Y,X,W
850 CALL VCHAR(Y,X,153,W)
860 NEXT J
```

```
870 READ I
880 FOR J=1 TO I
890 READ Y,X
900 CALL HCHAR(Y,X,136)
910 NEXT J
920 REM VARIABELN FESTLEGEN
930 AY=10
940 CONTR1=2
950 CONTR2=1
960 MX=7
970 MY=2
980 R1=1
990 REM HAUPTPROGRAMM
1000 CALL KEY(1,A1,B1)
1010 CALL KEY(2,A2,B2)
1020 CALL HCHAR(MY,MX,128)
1030 IF B2<>0 THEN 1360
1040 IF B1=0 THEN 2740
1050 REM GEHEN
1060 CALL SOUND(-20,-3,3)
1070 IF A1<>2 THEN 1140
1080 CALL HCHAR(MY,MX,ZE)
1090 MX=MX-1
1100 IF MX>=1 THEN 1120
1110 MX=32
1120 CALL GCHAR(MY,MX,ZE)
1130 CALL HCHAR(MY,MX,128)
1140 IF A1<>3 THEN 1210
1150 CALL HCHAR(MY,MX,ZE)
1160 MX=MX+1
1170 IF MX<=32 THEN 1190
1180 MX=1
1190 CALL GCHAR(MY,MX,ZE)
1200 CALL HCHAR(MY,MX,128)
1210 IF A1<>12 THEN 1270
1220 IF ZE=32 THEN 1270
1230 CALL HCHAR(MY,MX,ZE)
1240 MY=MY-1
1250 CALL GCHAR(MY,MX,ZE)
1260 CALL HCHAR(MY,MX,128)
1270 IF A1<>14 THEN 1340
1280 CALL GCHAR(MY+2,MX,CO)
1290 IF CO=32 THEN 1340
1300 CALL HCHAR(MY,MX,ZE)
1310 MY=MY+1
1320 CALL GCHAR(MY,MX,ZE)
1330 CALL HCHAR(MY,MX,128)
1340 GOTO 2740
1350 REM *** SPRINGEN ***
1360 CALL SOUND(-40,330,3)
1370 IF A2=2 THEN 1410
1380 IF A2=3 THEN 1780
1390 IF (A2=12)+(CONTR1<>0)=
-2 THEN 2200
1400 GOTO 2740
1410 IF A1<>2 THEN 1440
1420 R=-1
1430 GOTO 1480
1440 IF A1<>3 THEN 1470
1450 R=1
1460 GOTO 1480
1470 R=0
1480 CALL HCHAR(MY,MX,ZE)
1490 MY=MY-1
1500 MX=MX+R
1510 IF MX>=1 THEN 1530
1520 MX=32
1530 IF MX<=32 THEN 1550
1540 MX=1
1550 CALL GCHAR(MY,MX,ZE)
1560 CALL HCHAR(MY,MX,128)
1570 CALL GCHAR(MY+1,MX,CO)
1580 IF CO<>32 THEN 2740
1590 CALL HCHAR(MY,MX,ZE)
1600 MY=MY+1
1610 MX=MX+R
1620 IF MX>=1 THEN 1640
1630 MX=32
1640 IF MX<=32 THEN 1660
1650 MX=1
1660 CALL GCHAR(MY,MX,ZE)
1670 CALL HCHAR(MY,MX,128)
1680 CALL GCHAR(MY+1,MX,CO)
1690 IF CO<>32 THEN 2740
1700 CALL HCHAR(MY,MX,ZE)
1710 MY=MY+1
1720 IF MY=23 THEN 3640
1730 CALL GCHAR(MY,MX,ZE)
1740 CALL HCHAR(MY,MX,128)
1750 CALL GCHAR(MY+1,MX,CO)
1760 IF CO<>32 THEN 2740
1770 GOTO 3400
1780 IF A1<>2 THEN 1810
1790 R=-1
1800 GOTO 1850
1810 IF A1<>3 THEN 1840
1820 R=1
1830 GOTO 1850
1840 R=0
1850 CALL HCHAR(MY,MX,ZE)
1860 MY=MY-1
1870 MX=MX+R
1880 IF MX>=1 THEN 1900
1890 MX=32
1900 IF MX<=32 THEN 1920
1910 MX=1
1920 CALL GCHAR(MY,MX,ZE)
1930 CALL HCHAR(MY,MX,128)
1940 CALL GCHAR(MY+1,MX,CO)
1950 IF CO<>32 THEN 2740
```

```
1960 CALL HCHAR(MY,MX,ZE)
1970 MY=MY+1
1980 MX=MX+R
1990 IF MX>=1 THEN 2010
2000 MX=32
2010 IF MX<=32 THEN 2030
2020 MX=1
2030 CALL GCHAR(MY,MX,ZE)
2040 CALL HCHAR(MY,MX,128)
2050 CALL GCHAR(MY+1,MX,CO)
2060 IF CO<>32 THEN 2740
2070 CALL HCHAR(MY,MX,ZE)
2080 MY=MY+1
2090 IF MY=23 THEN 3640
2100 MX=MX+R
2110 IF MX<=32 THEN 2130
2120 MX=1
2130 IF MX>=1 THEN 2150
2140 MX=32
2150 CALL GCHAR(MY,MX,ZE)
2160 CALL HCHAR(MY,MX,128)
2170 CALL GCHAR(MY+1,MX,CO)
2180 IF CO<>32 THEN 2740
2190 GOTO 3400
2200 IF A1<>2 THEN 2230
2210 R=-1
2220 GOTO 2270
2230 IF A1<>3 THEN 2260
2240 R=1
2250 GOTO 2270
2260 R=0
2270 CALL HCHAR(MY,MX,ZE)
2280 CONTR1=CONTR1-1
2290 MY=MY-1
2300 MX=MX+R
2310 IF MX>=1 THEN 2330
2320 MX=32
2330 IF MX<=32 THEN 2350
2340 MX=1
2350 CALL GCHAR(MY,MX,ZE)
2360 CALL HCHAR(MY,MX,128)
2370 CALL GCHAR(MY+1,MX,CO)
2380 IF CO<>32 THEN 2740
2390 CALL HCHAR(MY,MX,ZE)
2400 MX=MX+R
2410 IF MX<=32 THEN 2430
2420 MX=1
2430 IF MX>=1 THEN 2450
2440 MX=32
2450 CALL GCHAR(MY,MX,ZE)
2460 CALL HCHAR(MY,MX,128)
2470 CALL GCHAR(MY+1,MX,CO)
2480 IF CO<>32 THEN 2740
2490 CALL HCHAR(MY,MX,ZE)
2500 MY=MY+1
2510 MX=MX+R
2520 IF MX<=32 THEN 2540
2530 MX=1
2540 IF MX>=1 THEN 2560
2550 MX=32
2560 CALL GCHAR(MY,MX,ZE)
2570 CALL HCHAR(MY,MX,128)
2580 CALL GCHAR(MY+1,MX,CO)
2590 IF CO<>32 THEN 2740
2600 CALL HCHAR(MY,MX,ZE)
2610 MY=MY+1
2620 IF MY=23 THEN 3640
2630 MX=MX+R
2640 IF MX<=32 THEN 2660
2650 MX=1
2660 IF MX>=1 THEN 2680
2670 MX=32
2680 CALL GCHAR(MY,MX,ZE)
2690 CALL HCHAR(MY,MX,128)
2700 CALL GCHAR(MY+1,MX,CO)
2710 IF CO<>32 THEN 2740
2720 GOTO 3400
2730 REM ***BESTIMMUNG DES
ZEICHENS UNTER DEM MEANNCHEN
 ***
2740 CALL GCHAR(MY+1,MX,CO)
2750 IF CO<>32 THEN 2770
2760 GOTO 3400
2770 IF CO=136 THEN 2800
2780 IF CO=144 THEN 2910
2790 GOTO 3000
2800 CALL HCHAR(MY+1,MX,32)
2810 CALL SOUND(-100,-3,0)
2820 PZ=PZ+INT(RND*50)+100
2830 IF (CONT=0)+(PZ>5000)=-
2 THEN 4010
2840 IF PZ<PZC THEN 1000
2850 PZC=PZC+1000+PZC1
2860 PZC1=PZC1+100
2870 IF PZC1<=500 THEN 2900
2880 RA=RA+1
2890 PZC1=0
2900 ON CONTR4 GOTO 4510,453
0
2910 CALL HCHAR(MY,MX,ZE)
2920 MX=MX+R1
2930 IF MX>=1 THEN 2950
2940 MX=32
2950 IF MX<=32 THEN 2970
2960 MX=1
2970 CALL GCHAR(MY,MX,ZE)
2980 CALL HCHAR(MY,MX,128)
2990 GOTO 3000
3000 CALL HCHAR(AY,APA,32,3)
3010 IF CO<>152 THEN 3100
```

```
3020 CALL HCHAR(MY,MX,ZE)
3030 ON (R1/2+1.5)GOTO 3040,
3070
3040 MY=MY+1
3050 CALL HCHAR(MY,MX,128)
3060 GOTO 3100
3070 MY=MY-1
3080 CALL HCHAR(MY,MX,128)
3090 REM***AUFZUG***
3100 AY=AY-R1
3110 CALL HCHAR(AY,APA,152,3
)
3120 IF (AY<5)+(AY>22)=0 THE
N 3140
3130 R1=-R1
3140 FOR I=1 TO RA
3150 CALL HCHAR(INT(RND*24+1
),INT(RND*32+1),32)
3160 NEXT I
3170 REM *** BOMBE ***
3180 ON CONTR5+1 GOTO 3190,3
260,3310
3190 IF RND>0.02 THEN 1000
3200 CONTR5=1
3210 CALL SOUND(-100,440,0)
3220 BX=INT(RND*30+2)
3230 BY=INT(RND*22+2)
3240 CALL HCHAR(BY,BX,43)
3250 GOTO 1000
3260 IF RND>0.05 THEN 1000
3270 CONTR5=2
3280 CALL SOUND(-100,220,0)
3290 CALL COLOR(2,11,2)
3300 GOTO 1000
3310 CALL KEY(1,A1,B1)
3320 IF (A1=7)+((ABS(MY-BY)<
2)+(ABS(MX-BX)<2)=-2)<>-2 TH
EN 3370
3330 CONTR5=0
3340 CALL COLOR(2,10,2)
3350 CALL HCHAR(BY,BX,137)
3360 GOTO 4320
3370 IF RND>0.05 THEN 1000
3380 CALL COLOR(2,10,2)
3390 GOTO 4230
3400 REM *** FALLEN ***
3410 CALL HCHAR(MY,MX,ZE)
3420 CALL KEY(1,A1,B1)
3430 IF (A1=19)+(T=0)+(CONTR
2<>0)=-3 THEN 3610
3440 IF T=0 THEN 3550
3450 MY=MY+1
3460 IF MY=23 THEN 3640
3470 CALL GCHAR(MY,MX,ZE)
3480 CALL SOUND(-1000,340-MY
*3,5)
3490 CALL HCHAR(MY,MX,129)
3500 CALL GCHAR(MY+1,MX,CO)
3510 IF CO=32 THEN 3410
3520 CALL SOUND(-500,-4,0)
3530 T=0
3540 GOTO 1000
3550 MY=MY+1
3560 CALL SOUND(-1000,700-MY
*5,5)
3570 IF MY=23 THEN 3640
3580 CALL GCHAR(MY,MX,ZE)
3590 CALL HCHAR(MY,MX,128)
3600 GOTO 3410
3610 T=1
3620 CONTR2=CONTR2-1
3630 GOTO 3410
3640 AZ=AZ-1
3650 IF AZ=0 THEN 3740
3660 CALL CLEAR
3670 CALL SCREEN(5)
3680 PRINT "     SIE HABEN NO
CH";INT(AZ):"              MEANN
CHEAN' !":"                UND":"
     ";PZ;"PUNKTE"
3690 FOR I=1 TO 300
3700 NEXT I
3710 CALL CLEAR
3720 CALL SCREEN(2)
3730 ON CONTR4 GOTO 4510,453
0
3740 CALL CLEAR
3750 CALL SCREEN(5)
3760 PRINT "SIE HABEN";INT(P
Z);"PUNKTE"
3770 INPUT "NAME ?":NAME$
3780 FOR I=1 TO 20
3790 IF PZ<PUNKTE(I)THEN 380
0 ELSE 3810
3800 NEXT I
3810 FOR J=19 TO I STEP -1
3820 LIST$(J+1)=LIST$(J)
3830 PUNKTE(J+1)=PUNKTE(J)
3840 NEXT J
3850 LIST$(I)=NAME$
3860 PUNKTE(I)=PZ
3870 FOR I=1 TO 20
3880 PRINT LIST$(I),PUNKTE(I
)
3890 NEXT I
3900 RESTORE
3910 PRINT "WENN SIE DIE TAB
ELLE ":"SPEICHERN WOLLEN DAN
N":"DRUACKEN SIE DIE LEERTAS
TE!"
3920 CALL KEY(0,A1,B1)
3930 IF B1=0 THEN 3920
```

```
3940 IF A1<>32 THEN 4000
3950 OPEN #1:"CS1",SEQUENTIA
L,INTERNAL,OUTPUT,FIXED 192
3960 FOR I=1 TO 5
3970 PRINT #1:LIST$(I),PUNKT
E(I),LIST$(I+1),PUNKTE(I+1),
LIST$(I+2),PUNKTE(I+2),LIST$
(I+3),PUNKTE(I+3)
3980 NEXT I
3990 CLOSE #1
4000 GOTO 530
4010 CONT=1
4020 AZ=AZ+1
4030 CALL SCREEN(5)
4040 CALL CLEAR
4050 PRINT " SIE HABEN EIN F
REIMEANNCHEN"
4060 PRINT "SIE HABEN NUN";A
Z;"MEANNCHEN"
4070 FOR I=1 TO 800
4080 NEXT I
4090 CONTR4=2
4100 CALL CLEAR
4110 CALL SCREEN(2)
4120 ON CONTR4 GOTO 4510,453
0
4130 DATA 34,13,3,5,6,4,18,2
,5,3,4,5,16,1,5,27,5,6,20,5,
7,10,3,8,4,4,8,16,2,9,3,1,9,
19,3,9,24
4140 DATA 9,10,2,1,11,3,1,11
,11,2,11,16,3,12,2,1,12,22,6
,13,1,7,13,10,3,13,30,3,14,1
6,4
4150 DATA 15,26,4,16,4,6,16,
17,5,18,1,1,19,2,5,20,16,3,2
1,1,2,21,7,3,21,25,6
4160 DATA 23,1,7,23,11,2,23,
18,15
4170 DATA 2,3,22,7,18,20,13
4180 DATA 22,14,1,4,6,4,2,17
,5,2,20,6,3,4,7,4,14,7,2,4,1
0,3,17,9,4,6,16,2,10,19,4
4190 DATA 17,19,6,7,21,2,4,2
4,2,13,23,5,19,22,4,22,25,1,
4,27,1,6,29,3,13,26,2,16,28,
2
4200 DATA 19,28,2,10,31,3
4210 DATA 23,8,3,8,3,23,2,19
,3,3,5,13,5,8,6,10,9,23,10,9
,11,5,17,14,17,16,17,6,22,3,
23
4220 DATA 12,24,7,25,18,25,7
,27,9,27,23,28,5,31,18,32
4230 CALL HCHAR(BY,BX,42)
4240 CALL SOUND(-100,-5,0)
4250 CALL HCHAR(BY-1,BX-1,32
,3)
4260 CALL HCHAR(BY,BX-1,32)
4270 CALL HCHAR(BY,BX+1,32)
4280 CALL HCHAR(BY+1,BX-1,32
,3)
4290 CALL HCHAR(BY,BX,32)
4300 CONTR5=0
4310 IF ((ABS(BY-MY)<7)+(ABS
(BX-MX)<10))<>-2 THEN 1000 E
LSE 3640
4320 PZ=PZ+INT(RND*100)+200
4330 IF (CONT=0)+(PZ>5000)=-
2 THEN 4010
4340 IF PZ<PZC THEN 1000
4350 PZC=PZC+1000+PZC1
4360 PZC1=PZC1+100
4370 IF PZC1<=500 THEN
4400
4380 RA=RA+1
4390 PZC1=0
4400 RESTORE
4410 GOTO 680
4420 DATA 28,1,3,4,5,4,12,5,
4,21,11,5,9,3,6,6,2,6,13,3,8
,8,7,11,4,3,11,14,3,10,19,4,
12,22,11
4430 DATA 9,24,4,10,29,2,16,
29,2,21,27,4
4440 DATA 13,18,2,14,4,3,14,
10,4,13,14,1,14,15,1,15,16,1
,16,17,12,18,14,4,19,27,4,20
,5,2
4450 DATA 20,21,5,21,10,9,23
,4,29,2,7,17,13,17,4,7,21,21
,6,2,12,6,2,9,7,2,4,8,4,18,9
,5,9,10,8
4460 DATA 5,12,3,15,13,6,5,1
6,6,8,18,5,17,18,4,5,19,2,14
,19,2,21,21,2,17,22,3,8,23,4
,13,23,3
4470 DATA 17,26,6,8,28,4,13,
28,3,5,31,18
4480 DATA 33,3,5,3,6,4,14,4,
19,4,20,4,28,5,10,6,4,6,5,6,
14,7,25,8,10,10,20,11,7,11,8
,12,13
4490 DATA 12,25,12,31,13,17,
14,7,16,20,16,26,16,32,17,6,
19,32,20,4,20,22,21,15,23,12
,23,13,23,24
4500 DATA 19,29,9,26
4510 RESTORE
4520 GOTO 680
4530 RESTORE 4420
4540 GOTO 680
```

LOHN-ABRECHNUNG

```
100 ! *********************
110 ! *                   *
120 ! *                   *
130 ! *                   *
140 ! * COPYRIGHT 1984 BY *
150 ! *   GUENTER STAHL   *
160 ! *   BEARBEITET VOM  *
170 ! *      TEAM DES     *
180 ! *  AKTUELL VERLAGES *
190 ! *      MUENCHEN     *
200 ! * BENOETIGTE GERAETE *
210 ! * TI 99/4 A KONSOLE *
220 ! * TI EXTENDET BASIC *
230 ! * KASSETTENRECORDER *
        SCHNITTSTELLE &
240 ! *      DRUCKER      *
250 ! *    BENOETIGTER    *
260 ! *   SPEICHERPLATZ   *
270 ! *    6246  BYTES    *
280 ! *********************
290 CALL VCHAR(1,1,32,768)::
 PRINT TAB(7);"LOHNABRECHNUN
G" :: PRINT
300 PRINT "..EINGABE DER PER
S. DATEN"
310 PRINT
320 LINPUT "MONAT (text)..:"
:M$
330 INPUT "JAHR..........:":
JS
340 LINPUT "DATUM.........:"
:DT$
350 LINPUT "NAME,VORNAME..:"
:A$
360 LINPUT "STRASSE.......:"
:A1$
370 LINPUT "PLZ ORT.......:"
:A2$
380 LINPUT "BANK..........:"
:C$
390 LINPUT "KONTONR.......:"
:C1$
400 LINPUT "BANKL.ZAHL....:"
:C2$
410 CALL VCHAR(1,1,32,768)
420 PRINT "......EINGABE DAT
EN"
430 PRINT
440 INPUT "PERSONALNR.....:"
:PN1
450 INPUT "ABTEILUNGSNR...:"
:N1
460 INPUT "STEUERKLASSE...:"
:N2
470 INPUT "KINDERZAHL.....:"
:N3
480 PRINT "0=KEINE 1= EV 2=R
K"
490 INPUT "KONFESSION.....:"
:N4
500 PRINT "10=AOK 20=IKK 30=
 "
510 INPUT "KRANKENVERS....:"
:N5
520 CALL VCHAR(1,1,32,768)
530 PRINT "....EINGABE DER B
EZUEGE"
540 PRINT
550 INPUT "STUNDENLOHN....:"
:STL
560 INPUT "ARBEITSSTUNDEN :"
:AS1
570 INPUT "UEBERSTUNDEN...:"
:UE
580 INPUT "SONN+FEIERTAGE :"
:SF
590 INPUT "URLAUBSLOHN....:"
:UL
600 INPUT "URLAUBSGELD....:"
:UG
610 INPUT "SONSTIGE BEZ...:"
:S
620 CALL VCHAR(1,1,32,768)
630 PRINT "...EINGABE DER AB
ZUEGE"
640 PRINT
650 INPUT "LOHNSTEUER.....:"
:LS
660 INPUT "KIRCHENSTEUER..:"
:KS
670 INPUT "KRANKENVERS....:"
:KV
680 INPUT "RENTENVERS.....:"
:RV
690 INPUT "ARBEITSL.VERS..:"
:AV
700 INPUT "PERS.ABZUEGE...:"
:PA
710 ! BERECHNUNG DER BETRAEG
E
720 V=52 ! VERM.W.ANL.
730 LOHN=STL*AS1 ! GRUNDLOHN
740 USTD=STL*UE ! UEBERSTUND
EN
750 BR=(LOHN+USTD+SF+UL+UG+S
+V)! BRUTTOLOHN
760 A=(LS+KS+KV+RV+AV)!ABZUE
GE
770 A1=(A+PA)! GESAMT ABZUEG
E
780 NV=(BR-A)! NETTO VERDIEN
ST
```

```
790 AZ=(NV-PA)! AUSZAHLUNGSB
ETRAG
800 CALL VCHAR(1,1,32,768)
810 !ERSTELLEN DER ABRECHNUN
G
820 DISPLAY AT(1,1):"LOHN-UN
D GEHALTSABRECHNUNG"
830 DISPLAY AT(2,1):"NAME/ST
.KL./REL/VERS/MON/J"
840 DISPLAY AT(3,1):;A$;N2;N
4;N5;DT$
850 DISPLAY AT(4,16):"DM";TA
B(25);"DM"
860 DISPLAY AT(5,1):"GRUNDGE
HALT" :: DISPLAY AT(5,14):US
ING "####.##":LOHN

870 DISPLAY AT(6,1):"UEBERST
UNDEN" :: DISPLAY AT(6,14):U
SING "####.##":USTD
880 DISPLAY AT(7,1):"VERM.W.
ANL" :: DISPLAY AT(7,14):USI
NG "####.##":V
890 DISPLAY AT(8,1):"STEUERP
FL." :: DISPLAY AT(8,22):USI
NG "####.##":BR
900 DISPLAY AT(9,1):"SONN/FE
IERT." :: DISPLAY AT(9,14):U
SING "####.##":SF
910 DISPLAY AT(10,1):"BRUTTO
" :: DISPLAY AT(10,22):USING
 "####.##":BR
920 DISPLAY AT(11,1):"------
-----------------------"

930 DISPLAY AT(12,1):"ABZUEG
E"
940 DISPLAY AT(13,1):"LOHNST
EUER" :: DISPLAY AT(13,14):U
SING "####.##":LS
950 DISPLAY AT(14,1):"KIRCHE
NST." :: DISPLAY AT(14,14):U
SING "####.##":KS
960 DISPLAY AT(15,1):"KRANKE
NVERS.";:: DISPLAY AT(15,14)
:USING "####.##":KV
970 DISPLAY AT(16,1):"RENTEN
VERS." :: DISPLAY AT(16,14):
USING "####.##":RV
980 DISPLAY AT(17,1):"ARB.L.
VERS." :: DISPLAY AT(17,14):
USING "####.##":AV
990 DISPLAY AT(18,1):"SONST.
ABZ." :: DISPLAY AT(18,14):U
SING "####.##":PA :: DISPLAY
 AT(18,22):USING "####.##":A
1
1000 DISPLAY AT(19,21):"____
____"
1010 DISPLAY AT(21,1):"AUSZA
HLUNG";TAB(18);"DM" :: DISPL
AY AT(21,22):USING "####.##"
:AZ
1020 DISPLAY AT(22,21):"====
===="
1030 DISPLAY AT(24,1):"1=DRU
CK/2=NEU/3=BILD/4=END"
1040 CALL KEY(0,KEY,STATUS):
: IF STATUS=0 THEN 1040
1050 IF KEY<49 OR KEY>52 THE
N 1040
1060 ON KEY-48 GOTO 1080,280
,870,1070
1070 END
1080 DISPLAY AT(24,1)BEEP:"D
RUCKER ANGESCHLOSSEN (J/N)?"
1090 CALL KEY(0,KE,ST):: IF
ST=0 THEN 1090 :: IF KE<74 O
R KE>78 THEN 1090 :: IF KE=7
4 THEN 1100 ELSE 800
1100 OPEN #1:"RS232" :: DISP
LAY AT(24,1):"..DATEN WERDEN
 AUSGEDRUCKT.."
1110 PRINT #1:CHR$(14)
1120 PRINT #1:TAB(4);"Lohnab
rechnung";CHR$(15);TAB(40);"
Monat ";TAB(50);"Jahr "
1130 PRINT #1 :: PRINT #1:TA
B(14);A$;TAB(54);M$;TAB(66);
JS
1140 PRINT #1 :: GOSUB 1590
1150 PRINT #1:TAB(4);"Person
al";TAB(14);"Datum";TAB(32);
"Abtnr.";TAB(42);"St.";TAB(5
0);"Kinder";TAB(60);"Rel.";T
AB(70);"Vers.";TAB(6);"Nr";T
AB(42);"kl";TAB(51);"zahl"
1160 GOSUB 1590
1170 PRINT #1:TAB(4);PN1;TAB
(14);DT$;TAB(32);N1;TAB(42);
N2;TAB(50);N3;TAB(60);N4;TAB
(70);N5
1180 GOSUB 1590 :: PRINT #1
1190 PRINT #1:TAB(36);"Stund
en";TAB(48);"Stundenlohn";TA
B(63);"Brutto-Betrag"
1200 PRINT #1
1210 PRINT #1,USING 1600:AS1
,STL,LOHN
1220 PRINT #1,USING 1610:UE,
STL,USTD
1230 PRINT #1,USING 1710:V :
: PRINT #1,USING 1620:SF
```

```
1240 PRINT #1,USING 1630:UL
1250 PRINT #1,USING 1640:UG
:: PRINT #1,USING 1700:S
1260 FOR I=1 TO 4 :: PRINT #
1 :: NEXT I
1270 GOSUB 1590
1280 PRINT #1:TAB(62);"Gesam
t-Brutto   " :: PRINT #1
1290 PRINT #1,USING 1650:BR
1300 PRINT #1 :: GOSUB 1590
1310 PRINT #1:TAB(4);"Steuer
";TAB(14);"Lohn";TAB(24);"Ki
rchen";TAB(32);"Kranken";TAB
(40);"Renten";TAB(48);"Arbei
tsl";TAB(62);"Gesetzl.Abzüge
"
1320 PRINT #1:TAB(4);"Brutto
";TAB(13);"Steuer";TAB(24);"
steuer";TAB(32);"vers.";TAB(
40);"vers.";TAB(50);"vers."
1330 PRINT #1 :: PRINT #1,US
ING 1660:BR,LS,KS,KV,RV,AV,A
1340 PRINT #1 :: GOSUB 1590
1350 PRINT #1:TAB(62);"Netto
-Verdienst"
1360 PRINT #1 :: PRINT #1,US
ING 1670:NV
1370 PRINT #1
1380 PRINT #1:TAB(40);"-----
----------------------------
-------"
1390 PRINT #1:TAB(10);A$;TAB
(44);"Netto - Abzüge"
1400 PRINT #1 :: PRINT #1,US
ING 1680:PA
1410 PRINT #1:TAB(10);A1$
1420 PRINT #1
1430 PRINT #1 :: PRINT #1:TA
B(10);A2$
1440 PRINT #1
1450 FOR I=1 TO 4 :: PRINT #
1 :: NEXT I :: GOSUB 1590
1460 PRINT #1:TAB(60);"Ausza
hlungsbetrag"
1470 PRINT #1 :: PRINT #1,US
ING 1690:AZ
1480 PRINT #1:TAB(60);"=====
========"
1490 PRINT #1
1500 GOSUB 1590
1510 PRINT #1:TAB(4);"Konton
r.";TAB(30);"Bank";TAB(50);"
Blz"
1520 PRINT #1 :: PRINT #1:TA
B(4);C1$;TAB(30);C$;TAB(50);
C2$
1530 PRINT #1
1540 GOSUB 1590
1550 FOR I=1 TO 9 :: PRINT #
1 :: NEXT I
1560 CLOSE #1
1570 CALL VCHAR(1,1,32,68)
1580 GOTO 800
1590 FOR I=1 TO 80 :: PRINT
#1:"-";:: NEXT I :: RETURN
1600 IMAGE "...Arbeitslohn..
.......................###.##...
...####.##.........####.##"
1610 IMAGE "...Überstunden..
.......................###.##...
...####.##.........####.##"
1620 IMAGE "...Sonn-Feiertag
e.............................
....................####.##"
1630 IMAGE "...Urlaubslohn..
.............................
....................####.##"
1640 IMAGE "...Urlaubsgeld..
.............................
....................####.##"
1650 IMAGE "...............
.............................
...............DM..####.##"
1660 IMAGE "...####.##...###
.##....###.##...###.##..###.
##..###.##......DM..####.##"
1670 IMAGE "...............
.............................
...............DM..####.##"
1680 IMAGE "...............
........................Pers
önliche Abzüge..DM..####.##"
1690 IMAGE "...............
.............................
...............DM..####.##"
1700 IMAGE "...Sonstiges....
.............................
....................####.##"
1710 IMAGE "...Verm.W.Anlage
.............................
....................####.##"
```

LOHNABRECHNUNG

Ein Programm, das beim Tester schon den Eindruck eines professionellen Anwenderprogrammes erweckt. Schön ist auch, daß, um es auf einer Parallel-Schnittstelle laufen zu lassen, nur die Zeile 1100 geändert werden muß! Interessant wäre noch das Programm selbst so zu erweitern, daß man Daten auf Diskette speichern kann! Dies jedoch nur als Anregung für unsere Leser.

Sonderheft Nr 1/85 - DM 9,80/ÖS 80/SFR 9,80

Spezial-Ausgabe COMPUTER AKTUELL

COMPUTER TEST-JAHRBUCH

100 Seiten EINKAUFSBERATER

ALLES ÜBER COMPUTER

NEU

WAS SIE BEIM COMPUTER-KAUF BEACHTEN MÜSSEN

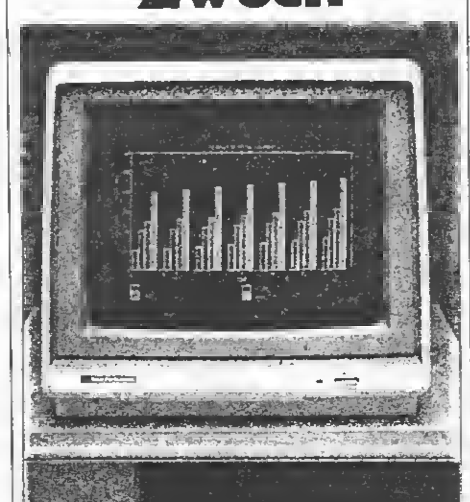
Die wichtigsten Computer im Test

Peripherie
Floppy
Drucker
Plotter

MARKTÜBERSICHT ALLE MSX-COMPUTER

Fachwort-lexikon

Welchen Computer zu welchem Zweck



Service-Themen:
Garantie
Bandsalat
Homecomputer
Personal-Computer

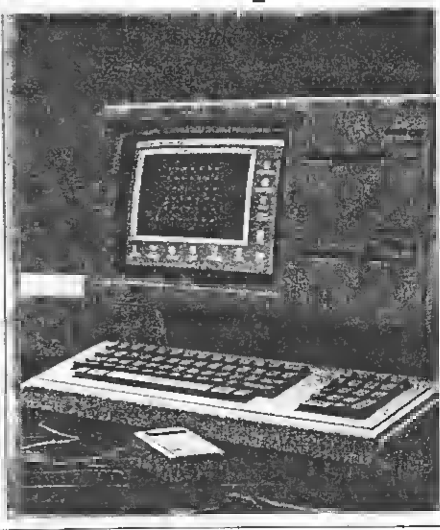
◊◊◊◊◊◊

JETZT AN IHREM KIOSK

◊◊◊◊◊◊

MSX — Die neue Generation!

12 Tips für den Kauf eines Homecomputers



DAS SUPER SERVICE PAKET! ABO- UND KASSETTENVERSAND

3

COUPON

Hiermit bestelle ich alle angebotenen Listings:
zum Super-Sonderpreis von DM 10,–
Den Betrag von DM 10,– habe ich ( ) in bar, ( ) als Scheck beigelegt

Meine Anschrift:
Name ____________
Vorname ____________
Straße ____________ Nr. ______
PLZ/Ort ____________
Unterschrift ____________

Ausschneiden und im Brief senden an:

AKTUELL TI–
KASSETTENSERVICE
POSTFACH 1107
8044 LOHHOF

Finden Sie Ihre TI REVUE nicht am Kiosk? Weil sie schon ausverkauft ist? Oder „Euer" Kiosk nicht beliefert wurde? Kein Problem! Für ganze 25 DM liefern wir Euch per Post sechs Hefte ins Haus (Ausland 30 DM). Einfach den Bestellschein unten ausschneiden – fotokopieren oder abschreiben, in einen Briefumschlag und ab per Post. TI REVUE kommt dann pünktlich ins Haus.

Wollt Ihr nicht mehr seitenlange Listings eintippen? Dann benutzt unseren Kassetten-Service: Für ganze zehn DM – nur Vorkasse, keine Rechnung – keine Nachnahme – erhalten Sie alle angebotenen Listings des jeweiligen Heftes auf Kassette per Post ins Haus. Achtung: Bestellungen ohne Geld werden nicht bearbeitet! Lieferzeit: In der Regel zwei bis drei Wochen! Wichtig: Unsere Kassetten werden digital in einem Profistudio von einem Masterband gezogen, nicht irgendwie per Recorder runtergenudelt. Selbstverständlich volles Umtauschrecht bei Nichtfunktionieren.

3

TI

Ich nehme zur Kenntnis, daß die Belieferung erst beginnt, wenn die Abo-Gebühr dem Verlag zugegangen ist.

TI AKTUELL
Abo-Service
Schellingstraße 33
8000 MÜNCHEN 40

Coupon

Ja, ich möchte von Ihrem Angebot Gebrauch machen.

Bitte senden Sie mir bis auf Widerruf ab sofort jeweils die nächsten sechs Ausgaben an untenstehende Anschrift.

Name ____________
Vorname ____________
Straße/Hausnr. ____________
Plz/Ort ____________

Ich bezahle:
- ☐ per beiliegendem Verrechnungsscheck
- ☐ gegen Rechnung
- ☐ bargeldlos per Bankeinzug von meinem Konto

bei (Bank) und Ort ____________
Kontonummer ____________
Bankleitzahl ____________
(steht auf jedem Kontoauszug)
Unterschrift ____________

Von meinem Widerspruchsrecht habe ich Kenntnis genommen

Unterschrift ____________

```
100 ! *********************
110 ! *                   *
120 ! * TI SENSO          *
130 ! *   Copyright by    *
140 ! *   Hans-J. Adler   *
150 ! *  Eagle-Computing  *
160 ! *                   *
170 ! *  Bearbeitet vom   *
180 ! *     Team des      *
190 ! * Aktuell Verlages  *
200 ! *     Muenchen      *
210 ! *                   *
220 ! *Benoetigte Geraete:*
230 ! *  TI99/4A Konsole  *
240 ! *     Ext. Basic    *
250 ! *                   *
260 ! * Speicherbelegung  *
270 ! *    6803 Bytes     *
280 ! *                   *
290 ! *********************

300 !
310 DATA 21,109,7,77,29,5,77
,197,11,133,109,4
320 DATA 2,139,4,175,1.5,175
,1.5,175,1.5,156,1.5,175,4,1
85,4,175
330 DATA 2,175,4,156,1.5,156
,1.5,156,1.5,139,1.5,156,4,1
75,4,139
340 DATA 2,139,4,175,1.5,175
,1.5,175,1.5,156,1.5,175,4,1
85,5,233
350 DATA 1.5,233,1.5,208,.2,
30000,1.5,208,1.5,185,1.5,17
5,1.5,156,1.5,139
360 DIM A(150),Z(4),S(4),F(4
),RN(6),R(6),L(32),K(32)
370 RANDOMIZE :: ON BREAK NE
XT
380 GOTO 420 :: CALL CLEAR :
: CALL MAGNIFY :: CALL SOUND
 :: CALL SPRITE
390 CALL KEY :: CALL JOYST :
: CALL CHAR :: CALL HCHAR ::
 CALL COLOR
400 CALL SCREEN :: CALL LOCA
TE :: CALL PATTERN :: CALL D
ELSPRITE
410 A$,I,KE,ST,SP,J,T,X,Y,SF
,N,HI,W :: !SP-
420 CALL CLEAR :: CALL MAGNI
FY(4)
430 CALL CHAR(59,"0008081",6
0,"3C4299A1A199423C",64,"")
440 CALL CHAR(91,"0044003844
447C44",92,"0044007C4444447C
",93,"0044004444444438")
450 A$="FFFF"
460 CALL CHAR(124,A$&RPT$("C
0",8)&A$&RPT$("0",8)&A$&RPT$
("03",8)&A$&RPT$("0",8))
470 CALL CHAR(132,"F88080F08
384FC03"&RPT$("0",48))
480 CALL CHAR(140,RPT$("AA55
",16))
490 A$=RPT$("F",8)&RPT$("F0"
,8)&RPT$("F",16)&RPT$("0F",8
)&RPT$("F",8)
500 CALL CHAR(136,A$,128,"00
FFFF",129,RPT$("F",16),130,"
",35,RPT$("F",16))
510 !  Felderpositionen
520 !  und Farben
530 FOR I=1 TO 4 :: READ Z(I
),S(I),F(I):: NEXT I
540 FOR I=1 TO 32 :: READ L(
I),K(I):: NEXT I
550 !
560 !  Titelbild  ++++++++
570 CALL COL(1,1,1):: J=0
580 FOR I=1 TO 6 :: CALL HCH
AR(I,1,129,32):: NEXT I
590 FOR I=7 TO 12 :: CALL HC
HAR(I,1,130,32):: NEXT I
600 FOR I=13 TO 18 :: CALL H
CHAR(I,1,35,32):: NEXT I
610 DISPLAY AT(3,8)SIZE(-12)
:"§T.I.-SENSO§"
620 DISPLAY AT(23,1)SIZE(-23
):"§<§EAGLE-computing§;83§"
630 CALL SCREEN(14):: CALL C
OLOR(1,10,7,13,5,8)
640 FOR I=2 TO 11 :: CALL CO
LOR(I,2,16):: NEXT I
650 CALL KEY(0,KE,ST):: IF S
T<=0 THEN J=J+1 :: IF J<500
THEN 650
660 !
670 !  Spielmenue  +++++++
680 CALL COL(2,4,14):: J=0
690 CALL P(3,5,"BITTE WÄHLEN
 SIE:"):: CALL P(4,1,RPT$(CH
R$(128),28))
700 CALL P(12,5,"1..SPIELANL
EITUNG"):: CALL P(14,5,"2..S
PIELWAHL")
710 CALL P(16,5,"3..ENDE")
720 FOR I=1 TO 32 :: CALL SO
UND(L(I)*100,K(I),5,K(I)*2+(
K(I)=3E4)*3E4,8)
```

```
730 CALL KEY(0,KE,ST):: IF K
E>48 AND KE<52 THEN 750
740 NEXT I :: CALL SOUND(800
,30000,30):: J=J+1 :: IF J=3
 THEN 550 ELSE 720
750 ON KE-48 GOTO 1340,760,1
210
760 !
770 !  Spielwahl  ++++++++
780 CALL COL(16,5,8)
790 FOR I=1 TO 13 :: CALL CO
LOR(I,16,5):: NEXT I
800 CALL P(3,9,"SPIELWAHL"):
: CALL P(4,1,RPT$(CHR$(128),
28))
810 CALL P(8,7,"1..LIMIT:..6
"):: CALL P(10,7,"2..LIMIT:
12")
820 CALL P(12,7,"3..LIMIT: 1
8"):: CALL P(14,7,"4..LIMIT:
 24")
830 CALL P(16,7,"5..LIMIT: 3
0"):: CALL P(18,7,"6..REKORD
VERSUCH")
840 CALL KEY(0,KE,ST):: IF K
E<49 OR KE>54 THEN 840
850 KE=KE-48 :: IF KE=6 THEN
 SP=150 ELSE SP=KE*6
860 !
870 !  Spielfeld  ++++++++
880 CALL CLEAR :: CALL SCREE
N(13)
890 FOR I=5 TO 8 :: CALL COL
OR(I,2,10):: NEXT I
900 CALL COLOR(1,2,16,2,2,10
,3,2,16,4,2,16,13,14,10)
910 FOR I=1 TO 4 :: CALL SPR
ITE(#I,140,F(I),Z(I),S(I))::
 NEXT I
920 CALL SPRITE(#5,132,5,9,1
4,#6,124,5,9,221)
930 A$=CHR$(129)&RPT$(CHR$(1
30),12)&CHR$(129)
940 CALL P(10,8,RPT$(CHR$(12
9),14)):: CALL P(11,8,A$)
950 CALL P(12,8,CHR$(129)&CH
R$(130)&"T.I.-SENSO"&CHR$(13
0)&CHR$(129))
960 CALL P(13,8,A$):: CALL P
(14,8,RPT$(CHR$(129),14))::
CALL WAIT(200)
970 !
980 !  Spielbeginn  ++++++
990 FOR I=1 TO SP :: A(I)=IN
T(RND*4)+1
1000 FOR J=1 TO I :: CALL LI
GHT(A(J)):: CALL WAIT(100)::
 NEXT J
1010 CALL SOUND(150,-1,0)
1020 FOR J=1 TO I :: T=0
1030 CALL JOYST(1,X,Y):: T=T
+1 :: IF T>50 THEN 1080
1040 W=ABS(X)+ABS(Y):: IF W=
0 OR W=8 THEN 1030
1050 IF X<0 THEN SF=2 ELSE I
F X>0 THEN SF=3 ELSE IF Y<0
THEN SF=4 ELSE SF=1
1060 CALL LIGHT(SF)
1070 IF A(J)=SF THEN 1140
1080 FOR N=1 TO 12 :: CALL P
ATTERN(#A(J),136):: CALL SOU
ND(50,220,0)
1090 CALL PATTERN(#A(J),140)
:: CALL SOUND(50,440,0):: NE
XT N
1100 IF SP<>150 THEN 1130
1110 IF I-1>=HI THEN CALL WA
IT(100):: CALL MEL :: HI=I-1
1120 GOTO 1200
1130 RN(SP/6)=RN(SP/6)+1 ::
GOTO 1200
1140 NEXT J
1150 DISPLAY AT(3,26)SIZE(4)
:USING "###":I :: CALL SOUND
(100,-3,0)
1160 IF I=10 THEN CALL LOCAT
E(#6,9,21[illegible])
1170 IF I=100 THEN CALL LOCA
TE(#6,9,213)
1180 CALL WAIT(250):: NEXT I
 :: CALL ROUND
1190 R(SP/6)=R(SP/6)+1
1200 CALL DELSPRITE(ALL):: G
OTO 1240
1210 !
1220 !  Spielende  ++++++++
1230 CALL CLEAR :: END
1240 !
1250 !  Tabelle  +++++++++
1260 CALL COL(2,10,7)
1270 CALL P(3,11,"TABELLE"):
: CALL P(4,1,RPT$(CHR$(128),
28))
1280 CALL P(7,15,"ja.....nei
n")
1290 FOR I=1 TO 5
1300 DISPLAY AT(7+2*I,3):USI
NG "LIMIT ##:..###.....###":
I*6,R(I),RN(I):: NEXT I
1310 CALL P(21,3,"(Spiel 6)"
)
```

```
1320 DISPLAY AT(23,3):USING
"HI-SCORE:..###":HI :: J=0
1330 J=J+1+(J=1)*2 :: CALL K
EY(J,KE,ST):: IF ST<=0 THEN
1330 ELSE 660
1340 !
1350 !  Spielanleitung  ++
1360 CALL COL(2,12,7)
1370 CALL P(2,7,"SPIELANLEIT
UNG"):: CALL P(3,1,RPT$(CHR$
(128),28))
1380 CALL P(5,1,"T.I.-SENSO
IST DIE ""TI-EX-..TENDED BAS
IC""-VERSION DES")
1390 CALL P(7,1,"BELIEBTEN S
PIELS ""SENSO"".")
1400 CALL P(9,1,"IHRE AUFGAB
E BESTEHT DARIN, EINE VOM CO
MPUTER VORGEGEBE-")
1410 CALL P(11,1,"NE FARBFOL
GE ZU WIEDERHOLEN.")
1420 CALL P(13,1,"DAS SPIELF
ELD ENTHÄLT VIER  GROSSE FAR
BFELDER, DIE BEI")
1430 CALL P(15,1,"BEDARF ""A
UFLEUCHTEN"" UND EINZUGEHÖRI
GES AKUSTISCHES")
1440 CALL P(17,1,"SIGNAL ABG
EBEN.")
1450 CALL P(19,1,"DER COMPUT
ER GIBT IHNEN EINE")
1460 CALL P(20,1,"IMMER LÄNG
ER WERDENDE FARB- FOLGE VOR,
 DIE SIE MIT HILFE")
1470 CALL P(22,1,"DES JOYSTI
CKS NR. 1 WIEDER- HOLEN MÜSS
EN.")
1480 CALL P(24,27,">>")
1490 CALL KEY(0,KE,ST):: IF
ST=0 THEN 1490 ELSE CALL CLE
AR
1500 CALL P(2,1,"UNTERLÄUFT
IHNEN EIN FEHLER BEI DER WIE
DERGABE ODER")
1510 CALL P(4,1,"ZÖGERN SIE
ZU LANGE, DANN...GILT DIE RU
NDE ALS VERLOREN.")
1520 CALL P(6,1,"EINE TABELL
E GIBT AUSKUNFT..ÜBER DIE BI
SHER ERZIELTEN")
1530 CALL P(8,1,"RESULTATE."
)
1540 CALL P(10,1,"SIE HABEN
DIE WAHL ZWISCHEN 6 VERSCHIE
DENEN SPIELSTUFEN:")
```

```
1550 CALL P(13,1,"BEI DEN ST
UFEN 1-5 WÄHLEN...SIE ALS ZI
EL EINE BEGRENZTE")
1560 CALL P(15,1,"ANZAHL VON
 WIEDERHOLUNGEN,..WÄHREND IH
NEN STUFE 6 DIE")
1570 CALL P(17,1,"MÖGLICHKEI
T EINES REKORDVER-SUCHES GIB
T !")
1580 CALL P(20,1,"***...SCHA
LTEN SIE DEN...******..""ALP
HA-LOCK"" AUS !..***")
1590 CALL P(23,6,"V.I.E.L...
S.P.A.S.S")
1600 CALL KEY(0,KE,ST):: IF
ST<=0 THEN 1600 ELSE 660
1610 !SP+
1620 !
1630 !
1640 !  +++++++++++++++++++
+++++  UNTERPROGRAMME  +++++
+++++++++++++++++++++
1650 !
1660 !  Aufleuchten  *****
1670 SUB LIGHT(X)
1680 CALL PATTERN(#X,136)::
CALL SOUND(200,110*X,2):: CA
LL WAIT(100)
1690 CALL PATTERN(#X,140)::
SUBEND
1700 !
1710 !  Zeitschleife  ****
1720 SUB WAIT(I)
1730 FOR X=1 TO I :: NEXT X
:: SUBEND
1740 !
1750 !  Display at  ******
1760 SUB P(X,Y,A$)
1770 DISPLAY AT(X,Y):A$ :: S
UBEND
1780 !
1790 !  Hi-Score-Melodie *
1800 SUB MEL
1810 FOR I=1 TO 150 STEP 3 :
: CALL SOUND(-100,110+I*5,2)
:: NEXT I :: SUBEND
1820 !
1830 !  Runde beendet  ***
1840 SUB ROUND
1850 FOR J=1 TO 3 :: FOR I=2
00 TO 400 STEP 50 :: CALL SO
UND(70,I,I/100):: NEXT I
1860 NEXT J :: FOR I=400 TO
200 STEP -40 :: CALL SOUND(7
0,I,I/100):: NEXT I
```

```
1870 CALL WAIT(400):: SUBEND
1880 !
1890 !  Farbdefinition  **
1900 SUB COL(X,Y,Z)
1910 CALL CLEAR :: CALL SCRE
EN(Z)
1920 FOR I=1 TO 13 :: CALL C
OLOR(I,X,Y):: NEXT I :: SUBE
ND
```

T.I. SENSO

Nachdem Sie eine der 6 Spielstufen gewählt haben, erscheinen auf dem Monitor 4 farbige Quadrate. Der Computer läßt nun eines dieser Quadrate aufleuchten und gibt Ihnen ein akustisches Signal. Jetzt müssen Sie per Joystick (1) das gleiche Feld zum Aufleuchten bringen. Haben Sie das richtige Feld angesteuert, bekommen Sie einen Punkt und der Computer läßt nun ein Feld mehr aufleuchten. Machen Sie einen Fehler oder zögern Sie zu lange, gilt die Runde für Sie als verloren. Die Anzahl Ihrer gewonnenen und verlorenen Runden wird in eine Liste eingetragen.
Sollten Sie den Rekordversuch in Stufe 6 überstehen, melden Sie sich beim Guiness-Buch an.

H.-J. Adler

Unser Aufruf uns Ihre Programme einzusenden, hatte einen unerwartet großen Erfolg. Uns liegen inzwischen mehrere hundert Programme vor. Fast alles sind Spiele. Von PAC MAN z.B. haben wir inzwischen mindestens 50 Versionen; wo bleiben die kommerziellen Anwendungen???
Übrigens, das Bundesligatabellenprogramm ist uns auch mindestens 10mal fast gleichzeitig eingeschickt worden. Da könnte fast der Verdacht aufkommen, daß alle es von der gleichen Quelle abgetippt haben.
Doch nun noch ein paar Bitten von uns: Gebt bitteschön nicht andere Aufträge mit zu den Programmsendungen. Diese gehen nämlich sofort zu unseren Programmierern, was zufolge hat, daß z.B. Abonnementaufträge, Leserbriefe oder Bestellungen für den Kassettenservice den Verlag erst mit zum Teil erheblicher Verzögerung erreichen. Diese Dinge bitte immer getrennt einsenden.

3-D IRRGARTEN

Ein hervorragendes Spiel für den TI99/4A und dazu noch in TI-Basic der Mehrzahl der Leser zugänglich.

Sie befinden sich in einem dreidimensionalen Irrgarten, dessen Ausdehnung Sie zu Beginn des Spiels durch Wahl einer der 6 Schwierigkeitsgrade bestimmen können. Die Laufrichtung geben Sie durch die Pfeiltasten an. Eine kleine Hilfe bietet Ihnen ein Pfeil am unteren Bildschirmrand, der die Himmelsrichtung des Ausganges anzeigt. Außerdem färbt sich der Boden des Labyrinthes gelb, wenn Sie den selben Gang ein zweites Mal betreten. Je höher Sie die Schwierigkeitsstufe wählen, desto mehr Gänge und Sackgassen hat das Labyrinth.
Vorsicht, manch einer hat sich schon hoffnungslos verirrt!

Dirk Gripp

3D-IRRGARTEN

```
100 REM********************
110 REM*                  *
120 REM*                  *
130 REM*   Copyright by   *
140 REM*    Dirk Gripp    *
150 REM*                  *
160 REM* Bearbeitet  vom  *
170 REM*     Team  des    *
180 REM* Aktuell Verlages *
190 REM*     Muenchen     *
200 REM*                  *
210 REM*Benoetigte Geraete*
220 REM* TI99/4A  Konsole *
230 REM*                  *
240 REM* Speicherbelegung *
250 REM*   10059 Bytes    *
260 REM********************
270 CALL SCREEN(1)
280 CALL CLEAR
290 PRINT "....SCHWIERIGKEIT
SGRADE:": : : :"1. DEMO-IRRG
.": :"2. KLEIN": :"3. MITTEL
": :"4. MITTEL + EINFACH": :
300 PRINT "5. SCHWER": :"6.
SCHWER + EINFACH": : : : :
310 FOR A=1 TO 8
320 CALL COLOR(A,4,1)
330 CALL SOUND(100,900,A*3.7
5)
340 NEXT A
350 FOR A=1 TO 4
360 READ Z,S
370 ZH1(A)=Z
380 SH1(A)=S
390 NEXT A
400 DATA 8,10,10,14,16,24,20
,30
410 CALL KEY(0,K,S)
420 IF S=0 THEN 410
430 IF (K<49)+(K>54)THEN 410
440 K=K-48
450 IF (K<>4)*(K<>6)THEN 470
460 LANG=6
470 IF (K=1)+(K=2)+(K=3)THEN
 490
480 K=INT(K/2+1.5)
490 ZH=ZH1(K)
500 SH=SH1(K)
510 CALL SOUND(800,300,0)
520 DIM FZ(40)
530 DIM FS(40)
540 DIM FELD$(2,30)
550 FOR A=2 TO SH-2
560 NULL$=NULL$&"0"
570 NEXT A
```

```
580 FOR A=3 TO ZH-1
590 FELD$(1,A)="0"&"1"&NULL$
&"1"
600 NEXT A
610 FOR A=1 TO SH
620 FELD$(1,2)=FELD$(1,2)&"1
"
630 FELD$(1,ZH)=FELD$(1,ZH)&
"1"
640 NEXT A
650 FELD$(1,1)=NULL$&"000"
660 REM
670 FOR S1=4 TO SH-2 STEP 2
680 FOR Z1=4 TO ZH-2 STEP 2
690 IF SEG$(FELD$(1,Z1),S1,1
)<>"1" THEN 730
700 NEXT Z1
710 NEXT S1
720 GOTO 1180
730 Z=Z1
740 S=S1
750 CALL SOUND(-4200,-8,0)
760 IF W=0 THEN 780
770 W=W+1
780 FZ(W)=Z
790 FS(W)=S
800 RANDOMIZE
810 L=INT((4+LANG)*RND)+1
820 IF L<=4 THEN 840
830 L=2
840 IF (L+2=AL)+(L-2=AL)THEN
 810
850 AL=L
860 W=W+1
870 GOSUB 890
880 GOTO 1000
890 SX=0
900 ZX=0
910 ON L GOTO 920,940,960,980
920 ZX=-1
930 RETURN
940 SX=1
950 RETURN
960 ZX=1
970 RETURN
980 SX=-1
990 RETURN
1000 FZ(W)=Z+ZX
1010 FS(W)=S+SX
1020 FOR GEF=0 TO W
1030 IF (FZ(W)+ZX=FZ(GEF))*(
FS(W)+SX=FS(GEF))THEN 1090
1040 NEXT GEF
1050 IF SEG$(FELD$(1,Z+2*ZX)
,S+2*SX,1)="1" THEN 1120
1060 Z=Z+2*ZX
1070 S=S+2*SX
1080 GOTO 760
1090 W=0
1100 Z1=Z1-2
1110 GOTO 700
1120 FOR F=0 TO W
1130 FELD$(1,FZ(F))=SEG$(FEL
D$(1,FZ(F)),1,FS(F)-1)&"1"&S
EG$(FELD$(1,FZ(F)),FS(F)+1,1
00)
1140 NEXT F
1150 CALL SOUND(-1200,S1*50+
110,0)
1160 W=0
1170 GOTO 700
1180 REM
1190 FELD$(1,ZH-1)=SEG$(FELD
$(1,ZH-1),1,SH-1)&"0"
1200 FOR A=1 TO ZH
1210 FOR B=1 TO SH
1220 FELD$(2,B)=FELD$(2,B)&S
EG$(FELD$(1,A),B,1)
1230 NEXT B
1240 CALL SOUND(100,(INT(A/2
)+1)*110,0)
1250 NEXT A
1260 ZEILE=3
1270 SPALTE=3
1280 L=2
1290 AL=2
1300 CALL CLEAR
1310 GOSUB 3760
1320 VOLL$="FFFFFFFFFFFFFFFF
"
1330 LEER$="0"
1340 HAVO$="3C3C3C3C3C3C3C3C
"
1350 HARE$="FF7F3F1F0F070301
"
1360 DIM ZL$(4)
1370 ZL$(1)="180C06FFFF060C1
8"
1380 ZL$(2)="183C7EDB9918181
8"
1390 ZL$(3)="183060FFFF60301
8"
1400 ZL$(4)="18181899DB7E3C1
8"
1410 HALI$="FFFEFCF8F0E0C080
"
1420 FOR A=1 TO 7
1430 CALL CHAR(A*8+25,VOLL$)
1440 CALL CHAR(A*8+26,LEER$)
1450 CALL CHAR(A*8+27,HALI$)
```

```
1460 NEXT A
1470 FOR A=8 TO 14
1480 CALL CHAR(A*8+25,VOLL$)
1490 CALL CHAR(A*8+26,LEER$)
1500 CALL CHAR(A*8+27,HARE$)
1510 NEXT A
1520 CALL CHAR(144,VOLL$)
1530 CALL CHAR(152,VOLL$)
1540 CALL CHAR(153,LEER$)
1550 FOR A=1 TO 16
1560 CALL COLOR(A,F1,F1)
1570 CALL SOUND(100,INT(10*R
ND)+110,A)
1580 NEXT A
1590 FOR A=1 TO 16
1600 READ S,ZE,O
1610 CALL VCHAR(1,S,ZE,O)
1620 NEXT A
1630 DATA 1,30,24,2,33,48,4,
41,96,8,49,72,11,57,48,13,65
,48,15,73,24,16,81,24,17,144
,24
1640 DATA 18,89,24,19,97,24,
20,105,48,22,113,48,24,121,7
2,27,129,96,31,137,48
1650 FOR A=1 TO 15
1660 READ Z,S,ZE,O
1670 CALL HCHAR(Z,S+1,ZE,O)
1680 NEXT A
1690 DATA 10,16,82,1,11,15,7
4,3,12,14,66,5,13,13,66,7,14
,12,58,9,15,11,58,11,16,10,5
0,13,17,9,50,15
1700 DATA 18,8,50,17,19,7,42
,19,20,6,42,21,21,5,42,23,22
,4,42,25,23,3,34,27,24,2,34,
29
1710 S=17
1720 FOR A=10 TO 24
1730 READ Z
1740 S=S-1
1750 CALL HCHAR(A,S,Z)
1760 NEXT A
1770 S=17
1780 FOR A=10 TO 24
1790 READ Z
1800 S=S+1
1810 CALL HCHAR(A,S,Z)
1820 NEXT A
1830 DATA 83,75,67,67,59,59,
51,51,51,43,43,43,43,35,35,9
1,99,107,107,115,115,123,123
,123
1840 DATA 131,131,131,131,13
9,139
1850 CALL HCHAR(23,17,143)
1860 GOTO 4070
1870 SR=0
1880 IF ZEILE<=10 THEN 1900
1890 IF ZEILE>ZH/2 THEN 1950
1900 IF SPALTE>SH/2 THEN 193
0
1910 GOSUB 3760
1920 GOTO 1990
1930 GOSUB 3800
1940 GOTO 1990
1950 IF SPALTE>SH/2 THEN 198
0
1960 GOSUB 3840
1970 GOTO 1990
1980 GOSUB 3880
1990 IF ZIEL<>1 THEN 2010
2000 F1=2
2010 CALL COLOR(15,F1,F1)
2020 CALL COLOR(16,F1,F1)
2030 CALL CHAR(144,VOLL$)
2040 FOR GZ=1 TO 7
2050 CALL COLOR(GZ,F1,F1)
2060 CALL COLOR(15-GZ,F1,F1)
2070 NEXT GZ
2080 GOSUB 3170
2090 FOR GZ=1 TO GHOE
2100 GANG(GZ)=GANG(GZ)+1
2110 NEXT GZ
2120 GOSUB 3170
2130 IF SEG$(FELD$(1,ZEILE+Z
X),SPALTE+SX,1)<>"2" THEN 21
50
2140 F4=12
2150 GOSUB 4320
2160 ZEUE=152
2170 FOR GG=1 TO GHOE
2180 ON GANG(GG)GOSUB 3300,3
320,3340,3370,3400,3440,3480
,3510,3530,3550,3580,3610,36
50,3690
2190 NEXT GG
2200 CALL COLOR(15,F1,F2)
2210 IF SGN(PO)<>-1 THEN 224
0
2220 FA=0
2230 GOTO 2250
2240 FA=PO
2250 FOR FAR=1 TO 7-FA
2260 CALL COLOR(FAR,F2,F4)
2270 CALL COLOR(15-FAR,F2,F4
)
2280 NEXT FAR
2290 CALL CHAR(143,ZL$(L))
2300 GOTO 2510
```

```
2310 REM
2320 CALL SOUND(-1,20000,30)
2330 CALL KEY(0,K,R)
2340 IF R=0 THEN 2330
2350 CALL SOUND(50,1000,10)
2360 METER1=METER1+1
2370 ZEILE=ZEILE+ZX
2380 SPALTE=SPALTE+SX
2390 IF SEG$(FELD$(1,ZEILE),
SPALTE,1)="2" THEN 2420
2400 METER2=METER2+1
2410 FELD$(1,ZEILE)=SEG$(FEL
D$(1,ZEILE),1,SPALTE-1)&"2"&
SEG$(FELD$(1,ZEILE),SPALTE+1
,51)
2420 IF F1=2 THEN 2460
2430 IF K=68 THEN 3920
2440 IF K=83 THEN 3970
2450 IF K=88 THEN 4020
2460 IF (F1=2)*(PO>5)THEN 46
50
2470 IF PO+1=7 THEN 4570
2480 CALL SOUND(1000,-4,0)
2490 SR=SR+1
2500 PO=PO+1
2510 IF (SR/2)-(INT(SR/2))=0
 THEN 2540
2520 CALL COLOR(16,F2,F1)
2530 GOTO 2550
2540 CALL COLOR(16,F1,F2)
2550 IF PO=0 THEN 2620
2560 IF SGN(PO)=-1 THEN 2600
2570 CALL COLOR(8-PO,F1,F1)
2580 CALL COLOR(PO+7,F1,F1)
2590 GOTO 2630
2600 CALL CHAR(144,HAVO$)
2610 GOTO 2630
2620 CALL CHAR(144,VOLL$)
2630 FOR GY=1 TO GHOE
2640 IF (GANG(GY)>7)+(GANG(G
Y)=-15)THEN 2700
2650 IF SGN(GANG(GY))=-1 THE
N 2680
2660 CALL COLOR(8-GANG(GY),F
4,F4)
2670 IF GANG(GY)-1=PO THEN 2
740
2680 CALL COLOR(9-ABS(GANG(G
Y)),F2,F4)
2690 GOTO 2740
2700 IF SGN(GANG(GY))=-1 THE
N 2730
2710 CALL COLOR(GANG(GY),F4,F4)
2720 IF GANG(GY)-8=PO THEN 2
740
2730 CALL COLOR(ABS(GANG(GY)
)-1,F2,F4)
2740 NEXT GY
2750 GOSUB 3170
2760 IF KOHOE=0 THEN 2970
2770 IF SR/2-INT(SR/2)=0 THE
N 2800
2780 ZEUE=153
2790 GOTO 2810
2800 ZEUE=152
2810 FOR GK=1 TO KOHOE
2820 GAKO(GK)=(ABS(GAKO(GK))
-1)*(SGN(GAKO(GK)))
2830 IF GAKO(GK)=8 THEN 2880
2840 IF GAKO(GK)<>-1 THEN 29
60
2850 GOSUB 3300
2860 CALL COLOR(7,F4,F4)
2870 GOTO 2900
2880 GOSUB 3510
2890 CALL COLOR(8,F4,F4)
2900 FOR B=GHOE TO 1 STEP -1
2910 GANG(B+1)=GANG(B)
2920 NEXT B
2930 GANG(1)=ABS(GAKO(GK))
2940 KOHOE=KOHOE-1
2950 GHOE=GHOE+1
2960 NEXT GK
2970 IF SGN(PO)=-1 THEN 3020
2980 FOR GW=1 TO GHOE
2990 IF (GANG(GW)-1<>PO)*(GA
NG(GW)-8<>PO)THEN 3010
3000 GOSUB 3720
3010 NEXT GW
3020 GOSUB 3040
3030 GOTO 2330
3040 REM
3050 IF (SR/2)-(INT(SR/2))=0
 THEN 3080
3060 ZEUE=152
3070 GOTO 3090
3080 ZEUE=153
3090 FOR GN=1 TO GHOE
3100 IF (GANG(GN)+1<>8)*(GAN
G(GN)+1<>15)THEN 3130
3110 GANG(GN)=(GANG(GN)+1)*(
-1)
3120 GOTO 3150
3130 GANG(GN)=GANG(GN)+1
3140 ON GANG(GN)GOSUB 3300,3
320,3340,3370,3400,3440,3480
,3510,3530,3550,3580,3610,36
50,3690
3150 NEXT GN
3160 RETURN
```

```
3170 REM
3180 FOR GN=1 TO GHOE
3190 IF (GANG(GN)=1)+(GANG(G
N)=8)THEN 3280
3200 IF SGN(GANG(GN))<>-1 TH
EN 3220
3210 GHOE=GHOE-1
3220 IF (GANG(GN)>7)+(GANG(G
N)=-15)THEN 3250
3230 ZEUE=(9-ABS(GANG(GN)))*
8+25
3240 GOTO 3260
3250 ZEUE=(ABS(GANG(GN))-1)*
8+25
3260 ON ABS(GANG(GN))-1 GOSU
B 3300,3320,3340,3370,3400,3
440,3480,3510,3530,3550,3580
,3610,3650,3690
3270 NEXT GN
3280 RETURN
3290 REM
3300 CALL VCHAR(1,16,ZEUE,9)
3310 RETURN
3320 CALL VCHAR(1,15,ZEUE,10
)
3330 RETURN
3340 CALL VCHAR(1,13,ZEUE,11
)
3350 CALL VCHAR(1,14,ZEUE,11
)
3360 RETURN
3370 CALL VCHAR(1,12,ZEUE,13
)
3380 CALL VCHAR(1,11,ZEUE,13
)
3390 RETURN
3400 FOR B=8 TO 10
3410 CALL VCHAR(1,B,ZEUE,15)
3420 NEXT B
3430 RETURN
3440 FOR B=4 TO 7
3450 CALL VCHAR(1,B,ZEUE,18)
3460 NEXT B
3470 RETURN
3480 CALL VCHAR(1,2,ZEUE,22)
3490 CALL VCHAR(1,3,ZEUE,22)
3500 RETURN
3510 CALL VCHAR(1,18,ZEUE,9)
3520 RETURN
3530 CALL VCHAR(1,19,ZEUE,10
)
3540 RETURN
3550 CALL VCHAR(1,20,ZEUE,11
)
3560 CALL VCHAR(1,21,ZEUE,11
)
3570 RETURN
3580 CALL VCHAR(1,22,ZEUE,13
)
3590 CALL VCHAR(1,23,ZEUE,13
)
3600 RETURN
3610 FOR B=24 TO 26
3620 CALL VCHAR(1,B,ZEUE,15)
3630 NEXT B
3640 RETURN
3650 FOR B=27 TO 30
3660 CALL VCHAR(1,B,ZEUE,18)
3670 NEXT B
3680 RETURN
3690 CALL VCHAR(1,31,ZEUE,22
)
3700 CALL VCHAR(1,32,ZEUE,22
)
3710 RETURN
3720 REM
3730 ZEUE=144
3740 ON GANG(GW)GOSUB 3300,3
320,3340,3370,3400,3440,3480
,3510,3530,3550,3580,3610,36
50,3690
3750 RETURN
3760 F1=3
3770 F2=4
3780 F4=15
3790 RETURN
3800 F1=5
3810 F2=6
3820 F4=16
3830 RETURN
3840 F1=5
3850 F2=8
3860 F4=15
3870 RETURN
3880 F1=9
3890 F2=10
3900 F4=16
3910 RETURN
3920 IF L=4 THEN 3950
3930 L=L+1
3940 GOTO 4070
3950 L=1
3960 GOTO 4070
3970 IF L=1 THEN 4000
3980 L=L-1
3990 GOTO 4070
4000 L=4
4010 GOTO 4070
4020 IF L=2 THEN 4000
4030 IF L=1 THEN 4060
4040 L=L-2
```

```
4050 GOTO 4070
4060 L=3
4070 GOSUB 890
4080 IF ZX=0 THEN 4130
4090 SZ=2
4100 SUFE=SPALTE
4110 SUAE=ZEILE
4120 GOTO 4160
4130 SZ=1
4140 SUFE=ZEILE
4150 SUAE=SPALTE
4160 REM
4170 IF SEG$(FELD$(SZ,SUFE),
SUAE+ZX+SX,1)<>"1" THEN 4190
4180 IF (ZEILE=3)*(SPALTE=3)
*(L=2)THEN 3920 ELSE 4570
4190 ALSX=SX
4200 AL=L
4210 ALZX=ZX
4220 PO1=0
4230 SUAE1=SUAE
4240 SUAE1=SUAE1+ZX+SX
4250 IF SEG$(FELD$(SZ,SUFE),
SUAE1,1)="1" THEN 4300
4260 PO1=PO1+1
4270 IF (SUFE=ZEILE)*(SUAE1=
SH)*(ZEILE=ZH-1)THEN 4290
4280 GOTO 4240
4290 ZIEL=1
4300 PO=7-PO1
4310 GOTO 1870
4320 REM
4330 GHOE=0
4340 IF PO<=0 THEN 4350 ELSE
 4360
4350 PO1=7
4360 FOR A=0 TO PO1-1
4370 SUGA$=SEG$(FELD$(SZ,SUF
E+ZX+SX),SUAE+(PO1-A)*(SX+ZX
),1)
4380 IF (SUGA$<>"0")*(SUGA$<
>"2")THEN 4400
4390 ON L GOSUB 4790,4760,47
90,4760
4400 SUGA$=SEG$(FELD$(SZ,SUF
E-(ZX+SX)),SUAE+(PO1-A)*(SX+
ZX),1)
4410 IF (SUGA$<>"0")*(SUGA$<
>"2")THEN 4430
4420 ON L GOSUB 4760,4790,47
60,4790
4430 NEXT A
4440 REM
4450 IF SGN(PO)<>-1 THEN 455
0
4460 KOHOE=0
4470 FOR A=ABS(PO)+8 TO 8 ST
EP -1
4480 SUGA$=SEG$(FELD$(SZ,SUF
E+ZX+SX),SUAE+A*(SX+ZX),1)
4490 IF (SUGA$<>"0")*(SUGA$<
>"2")THEN 4510
4500 ON L GOSUB 4850,4820,48
50,4820
4510 SUGA$=SEG$(FELD$(SZ,SUF
E-(ZX+SX)),SUAE+A*(ZX+SX),1)
4520 IF (SUGA$<>"0")*(SUGA$<
>"2")THEN 4540

4530 ON L GOSUB 4820,4850,48
20,4850
4540 NEXT A
4550 RETURN
4560 REM
4570 CALL SOUND(-400,110,0,-
6,10)
4580 ZX=ALZX
4590 SX=ALSX
4600 METER1=METER1-1
4610 L=AL

4620 SPALTE=SPALTE-SX
4630 ZEILE=ZEILE-ZX
4640 GOTO 2330
4650 REM
4660 FOR A=2 TO 16
4670 CALL SCREEN(A)
4680 CALL SOUND(100,INT((100
0-110+1)*RND)+110,0)
4690 NEXT A
4700 CALL SCREEN(1)
4710 CALL KEY(0,Z,X)
4720 IF X=0 THEN 4710
4730 CALL CLEAR
4740 PRINT "GELAUFENE METER:
 ";METER1: : : :"MINDEST-MET
ER";METER2

4750 END
4760 GHOE=GHOE+1
4770 GANG(GHOE)=15-PO1+A
4780 RETURN
4790 GHOE=GHOE+1
4800 GANG(GHOE)=8-PO1+A
4810 RETURN
4820 KOHOE=KOHOE+1
4830 GAKO(KOHOE)=A+2
4840 RETURN
4850 KOHOE=KOHOE+1
4860 GAKO(KOHOE)=(A-5)*-1
4870 RETURN
```

HAUS/MO

Dieses Programm gehört zu einem Paket von 3 Programmen, ist aber auch alleine lauffähig.
Die Arbeitsweise des Programmes ist schnell und genau.
Es schafft Ordnung in allen monatlichen Geldbewegungen, ohne lange in Unterlagen und Belegen herumzuwühlen.

HAUS/JA

Eine gute Erweiterung des obigen Programmes. Ist aber nur mit den Programmen Haus/Mo und Haus/Gen zusammen lauffähig.
Es wertet die Daten aus, die mit Haus/Mo erstellt wurden.

HAUS/GEN

Das Kurzprogramm Haus/Gen erstellt lediglich die Dateien, die für das Programm Haus/Mo erforderlich sind. Nach dem Erstellen der Dateien ist das Programm Haus/Gen wieder zu löschen und Haus/Ja zu laden.

HAUS/MO

Nach Start des Programmes bekommen Sie auf der linken Seite des Bildschirmes 20 mögliche Ausgabekriterien zur Wahl gestellt, auf der rechten Seite 8 Einnahmekriterien. Alle Kriterien sind durchnumeriert und durch Wahl der Kennziffer haben Sie nun die Möglichkeit, Einnahmen, Ausgaben oder Umbuchungen einzugeben. Die Summen werden getrennt nach der jeweiligen Kategorie von den Einnahmen abgezogen. Haben Sie alle Geldbewegungen des Tages eingegeben, können Sie nun die Auswertung im Menü anwählen. Sie bekommen nun eine genaue Übersicht über Ihre Kontostände und die einzelnen Umbuchungen und Ausgaben.
Anschließend können Sie die Daten auf Diskette abspeichern, so daß sie Ihnen am Tag danach wieder zur Verfügung stehen. Am Ende eines Monats wählen Sie die Funktion 'Monat abschließen'. Hierauf werden die Dateien für die Weiterverarbeitung mit dem Programm 'Haus/Ja' vorbereitet und die Daten im Haus/Mo auf '0' gesetzt, so daß Ihnen das Programm im darauffolgenden Monat wieder zur Verfügung steht.

Steuerung:
Menüwahl durch Eingabe der Kennziffer.
Wahl der Ausgaben durch Kennziffer.
Wahl der Einnahmen durch Kennziffer.
Taste 'B' Kassenstand (bar)
Taste 'G' Girokonto
Taste 'P' Postgirokonto
Taste 'S' Sparkonto
Taste 'K' Kreditkonto
Taste '0' Abbuchung oder Neuverbuchung der Einnahmen.

Alfred Nitschke

HAUSHALT

```
100 ! *********************
110 ! * HAUSHALT           *
120 ! *                    *
130 ! *   Copyright by     *
140 ! *    A. Nitschke     *
150 ! *  Bearbeitet vom    *
160 ! *     Team des       *
170 ! * Aktuell Verlages   *
180 ! *     Muenchen       *
190 ! *                    *
200 ! *Benoetigte Geraete:*
210 ! *  TI99/4A Konsole   *
220 ! *    Ext. Basic      *
230 ! *Disk od. Cass.Rec.  *
240 ! *  moegl. Drucker    *
250 ! *                    *
260 ! * Speicherbelegung:  *
270 ! *    7819 Bytes      *
280 ! *                    *
290 ! *********************
300 ! MONATLICHE AUSGABEN UN
D EINNAHMEN
310 ! *********************
***********
320 CALL CLEAR :: CALL SCREE
N(11)
330 DISPLAY AT(2,1):RPT$("*"
,28):"**..MONATLICHE..AUSGAB
EN..**":"**..UND..EINNAHMEN.
.......**":RPT$("*",28)
340 CALL CHAR(130,"080808080
8080808")
350 CALL CHAR(131,"00000000F
F")
360 DIM B(20),Z(20),SB(20)
370 DISPLAY AT(10,1):"WAHL-L
ISTE" :: DISPLAY AT(13,3):"*
1*..EINLESEN BISH. DATEN" ::
 DISPLAY AT(14,3):"*2*..EING
ABE:..AUSGABEN"
380 DISPLAY AT(15,3):"*3*..E
INGABE:..EINNAHMEN" :: DISPL
AY AT(16,3):"*4*..AUSWERTUNG
" :: DISPLAY AT(17,3):"*5*..
DATEN SPEICHERN"
390 DISPLAY AT(18,3):"*6*..M
ONAT BEENDEN" :: DISPLAY AT(
19,3):"*7*..AUSDRUCKEN" :: D
ISPLAY AT(20,3):"*8*..ENDE"
400 DISPLAY AT(23,1)BEEP:"KE
NNUMMER EINGEBEN"
410 CALL KEY(0,K,S)
420 IF (S=0)+(K<48)+(K>56)<>
0 THEN 410
430 ON K-48 GOTO 440,820,115
0,1450,1730,1910,2050,2040
```

```
440 ! LESEN DER BISH.SUMM.
450 ON ERROR 570
460 OPEN #20:"DSK1.HAUS/1",I
NTERNAL,INPUT ,FIXED 255
470 FOR F=1 TO 20
480 INPUT #20:SB(F),
490 NEXT F
500 FOR E=1 TO 8
510 INPUT #20:SZ(E),
520 NEXT E
530 CLOSE #20
540 OPEN #21:"DSK1.HAUS/2",I
NTERNAL,INPUT ,FIXED 255
550 INPUT #21:KS,KG,KP,KZ,KK
,U,SE,SU,DATUM$
560 CLOSE #21
570 IF KG=0 AND KP=0 THEN 58
0 ELSE 650
580 DISPLAY AT(4,1)ERASE ALL
:"KONTO-STAENDE EINGEBEN" ::
 DISPLAY AT(8,1):"KASSE /BAR
": :"GIROKONTO": :"POSTSCHEC
KKONTO": :"SPARKONTO": :"KRE
DITKONTO"
590 ACCEPT AT(8,18)VALIDATE(
DIGIT,"."):KS
600 ACCEPT AT(10,18)VALIDATE
(DIGIT,"-."):KG
610 ACCEPT AT(12,18)VALIDATE
(DIGIT,"-."):KP
620 ACCEPT AT(14,18)VALIDATE
(DIGIT,"."):KZ
630 ACCEPT AT(16,18)VALIDATE
(DIGIT,"-."):KK
640 GOTO 320
650 DISPLAY AT(1,6)ERASE ALL
:"* KONTO-STAND *"
660 DISPLAY AT(3,6):"DATUM:
";DATUM$
670 DISPLAY AT(6,1):"KASSENB
ESTAND...:" :: DISPLAY AT(6,
20):USING "######.##":KS
680 DISPLAY AT(8,1):"GIROKON
TO.......:" :: DISPLAY AT(8,
20):USING "######.##":KG
690 DISPLAY AT(10,1):"POSTSC
HECKKONTO :" :: DISPLAY AT(1
0,20):USING "######.##":KP
700 DISPLAY AT(12,1):"SPARKO
NTO.......:" :: DISPLAY AT(1
2,20):USING "######.##":KZ
710 DISPLAY AT(14,1):"KREDIT
KONTO.....:" :: DISPLAY AT(1
4,20):USING "######.##":KK
720 DISPLAY AT(18,1):"SUMME
EINNAHMEN :" :: DISPLAY AT(1
8,20):USING "######.##":SE
730 DISPLAY AT(20,1):"SUMME
AUSGABEN..:" :: DISPLAY AT(2
0,20):USING "######.##":SU
740 DISPLAY AT(22,1):"UMBUCH
UNGEN.....:" :: DISPLAY AT(2
2,20):USING "######.##":U
750 !
760 !
780 CALL SOUND(150,1400,5)
790 CALL KEY(0,K,S):: IF S=0
 THEN 790
800 GOTO 320
810 !
820 ! EINGEBEN WEITERER
      AUSGABEN
830 CALL CLEAR
840 DISPLAY AT(1,1):"AUSGABE
N"
850 RESTORE 1500
860 FOR I=1 TO 20 :: READ A$
 :: DISPLAY AT(I+2,1):USING
"##":I :: DISPLAY AT(I+2,4)S
IZE(12):A$ :: NEXT I
870 CALL VCHAR(1,19,130,22)
880 DISPLAY AT(1,19):"AUSGAB
EN" :: DISPLAY AT(2,19):"BIS
HER:"
890 DISPLAY AT(4,21):USING "
#####.##":SU
900 DISPLAY AT(7,19):"ART:..
0"
910 ACCEPT AT(7,25)BEEP VALI
DATE(DIGIT)SIZE(-2):A :: DIS
PLAY AT(22,19):""
920 IF A=0 THEN 150
930 IF A>20 THEN 900
940 DISPLAY AT(10,19):"BETRA
G:" :: ACCEPT AT(12,21)BEEP
VALIDATE(DIGIT,"."):B(A)
950 DISPLAY AT(15,19):"VON K
TO..B" :: ACCEPT AT(15,28)BE
EP VALIDATE("B,P,G,S")SIZE(-
1):K$
960 DISPLAY AT(17,19):"AUF K
TO..0"
970 IF K$="B" THEN 1010
980 IF K$="G" THEN 1020
990 IF K$="P" THEN 1030
1000 IF K$="S" THEN 1040
1010 ACCEPT AT(17,28)BEEP VA
LIDATE("G,P,S,K,0")SIZE(-1):
T$ :: KS=KS-B(A):: SB(A)=SB(
A)+B(A):: GOTO 1050
1020 ACCEPT AT(17,28)BEEP VA
LIDATE("B,P,S,K,0")SIZE(-1):
T$ :: KG=KG-B(A):: SB(A)=SB(
A)+B(A):: GOTO 1050
```

```
1030 ACCEPT AT(17,28)BEEP VA
LIDATE("B,G,S,K,O")SIZE(-1):
T$ :: KP=KP-B(A):: SB(A)=SB(
A)+B(A):: GOTO 1050
1040 ACCEPT AT(17,28)BEEP VA
LIDATE("B,G,P,K,O")SIZE(-1):
T$ :: KZ=KZ-B(A):: SB(A)=SB(
A)+B(A)
1050 IF T$="G" THEN KG=KG+B(
A)
1060 IF T$="B" THEN KS=KS+B(
A)
1070 IF T$="P" THEN KP=KP+B(
A)
1080 IF T$="S" THEN KZ=KZ+B(
A)
1090 IF T$="K" THEN KK=KK+B(
A)
1100 IF T$="K" OR T$="S" OR
T$="O" THEN SU=SU+B(A)ELSE 1
120
1110 GOTO 1130
1120 U=U+B(A):: DISPLAY AT(2
2,19):"UMBUCHUNG"
1130 DISPLAY AT(7,25):"" ::
DISPLAY AT(12,21):"" :: DISP
LAY AT(15,28):"" :: DISPLAY
AT(17,28):""
1140 GOTO 880
1150 !  ERFASSUNG WEITERER
        EINNAHMEN
1160 CALL CLEAR
1170 DISPLAY AT(1,1):"EINNAH
MEN"
1180 RESTORE 1620 :: FOR A=1
 TO 8 :: READ EI$ :: DISPLAY
 AT(5+A,1):STR$(A):: DISPLAY
 AT(5+A,3):EI$ :: NEXT A
1190 CALL VCHAR(1,19,130,20)
1200 DISPLAY AT(1,19):"EINNA
HMEN" :: DISPLAY AT(2,19):"B
ISHER:"
1210 DISPLAY AT(4,21):USING
"#####.##":SE
1220 DISPLAY AT(7,19):"ART:.
.O" :: ACCEPT AT(7,25)BEEP V
ALIDATE(DIGIT)SIZE(-1):E
1230 IF E=0 THEN 150
1240 IF E>8 THEN 1220
1250 DISPLAY AT(10,19):"BETR
AG:" :: ACCEPT AT(12,21)BEEP
 VALIDATE(DIGIT,"."):Z(E)
1260 DISPLAY AT(15,19):"VON
KTO..O" :: ACCEPT AT(15,28)B
EEP VALIDATE("G,P,S,O")SIZE(
-1):K$
1270 DISPLAY AT(17,19):"AUF
KTO..B" :: DISPLAY AT(22,19)
:""
1280 IF K$="G" THEN 1320 ELS
E IF K$="P" THEN 1330 ELSE I
F K$="S" THEN 1340 ELSE IF K
$="O" THEN 1350
1320 ACCEPT AT(17,28)BEEP VA
LIDATE("B,P,S")SIZE(-1):T$ :
: KG=KG-Z(E):: SZ(E)=SZ(E)+Z
(E):: GOTO 1380
1330 ACCEPT AT(17,28)BEEP VA
LIDATE("B,G,S")SIZE(-1):T$ :
: KP=KP-Z(E):: SZ(E)=SZ(E)+Z
(E):: GOTO 1380
1340 ACCEPT AT(17,28)BEEP VA
LIDATE("B,G,P")SIZE(-1):T$ :
: KZ=KZ-Z(E):: SZ(E)=SZ(E)+Z
(E):: GOTO 1380
1350 ACCEPT AT(17,28)BEEP VA
LIDATE("B,G,P,S,K")SIZE(-1):
T$
1360 IF T$="K" THEN KK=KK+Z(
E):: GOTO 1440
1370 SZ(E)=SZ(E)+Z(E):: SE=S
E+Z(E)
1380 IF T$="G" THEN KG=KG+Z(
E)
1390 IF T$="B" THEN KS=KS+Z(
E)
1400 IF T$="P" THEN KP=KP+Z(
E)
1410 IF T$="S" THEN KZ=KZ+Z(
E)
1420 IF K$="O" THEN 1440
1430 U=U+Z(E):: DISPLAY AT(2
2,19):"UMBUCHUNG" ::
1440 DISPLAY AT(7,25):"" ::
DISPLAY AT(12,21):"" :: DISP
LAY AT(15,28):"" :: DISPLAY
AT(17,28):"" :: GOTO
1210
1450 ! AUSWERTUNG
1460 CALL CLEAR
1470 GS=0
1480 RESTORE 1500
1490 FOR F=1 TO 20
1500 DATA HAUSHALT,MIETE/NK,
EINRICHTUNG,KLEIDUNG,AUTO,VE
RSICHERUNG,AUSGEHEN,CASINO
1510 DATA TELEFON,URLAUB,FS/
RADIO,ZEITSCHRIFTEN,COMPUTER
,SCHALLPLATTEN,VIDEO,WEITERB
ILDUNG,TOTO/LOTTO,KREDIT,UEB
ERWEISUNG,SONSTIGES
1520 READ AUSG$
```

```
1530 DISPLAY AT(F,1)SIZE(5):
AUSG$ :: DISPLAY AT(F,7):USI
NG "####.##":SB(F)
1540 GS=GS+SB(F)
1550 NEXT F
1560 DISPLAY AT(22,1):"SUMME
" :: DISPLAY AT(22,6):USING
"#####.##":GS
1570 CALL VCHAR(1,16,130,23)
1580 DISPLAY AT(1,19):"AUSWE
RTUNG" :: DISPLAY AT(2,19):S
EG$(DATUM$,1,6)&"19"&SEG$(DA
TUM$,7,2)
1590 GZ=0
1600 RESTORE 1620
1610 FOR E=1 TO 8
1620 DATA GEHALT,TR-GELD,VER
SICHERUNG,ZINSEN,GUTSCHRIFTE
N,VERKAEUFE,TOTO/LOTTO,SONST
IGES
1630 READ EINN$
1640 DISPLAY AT(E+5,15)SIZE(
6):EINN$ :: DISPLAY AT(E+5,2
2):USING "####.##":SZ(E)
1650 GZ=GZ+SZ(E)
1660 NEXT E
1670 DISPLAY AT(15,15):"SUMM
E:" :: DISPLAY AT(15,20):USI
NG "######.##":GZ
1680 DISPLAY AT(23,1)SIZE(5)
:"REAL:" :: DISPLAY AT(23,6)
:USING "#####.##":SU :: DISP
LAY AT(16,15):"REAL:" :: DIS
PLAY AT(16,21):USING "#####.
##":SE
1690 IF SE>SU THEN 1700 ELSE
 1710
1700 DISPLAY AT(20,15)BEEP:"
EINNAHMEN-PLUS" :: DISPLAY A
T(21,21):USING "#####.##":SE
-SU :: GOTO 1720
1710 DISPLAY AT(20,15)BEEP:"
AUSG. ZU HOCH" :: DISPLAY AT
(21,21):USING "#####.##":SU-
SE
1720 CALL KEY(0,K,S):: IF S=
0 THEN 1720 :: CALL CLEAR ::
 GOTO 670
1730 ! SPEICHERN DER ZW.SUMMEN
1740 CALL CLEAR
1750 DISPLAY AT(1,1):RPT$("-
",28):"***..DATEN..ABSPEICHE
RN..***":RPT$("-",28)
1760 DISPLAY AT(8,1):"LETZTE
 EINGABE:" :: DISPLAY AT(8,1
7):DATUM$
1770 DISPLAY AT(12,1):"HEUTI
GES DATUM:" :: DISPLAY AT(12
,17):DATUM$ :: ACCEPT AT(12,
17)SIZE(-8):DATUM$
1780 OPEN #20:"DSK1.HAUS/1",
INTERNAL,OUTPUT,FIXED 255
1790 DISPLAY AT(1,1):RPT$("-
",28):"* DATEN WERDEN GESPEI
CHERT *":RPT$("-",28)
1800 FOR F=1 TO 20
1810 PRINT #20:SB(F),
1820 NEXT F
1830 FOR E=1 TO 8
1840 PRINT #20:SZ(E),
1850 NEXT E
1860 CLOSE #20
1870 OPEN #21:"DSK1.HAUS/2",
INTERNAL,OUTPUT,FIXED 255
1880 PRINT #21:KS,KG,KP,KZ,K
K,U,SE,SU,DATUM$
1890 CLOSE #21
1900 GOTO 150
1910 CALL CLEAR !MONAT BEEND
EN
1920 DISPLAY AT(3,1):RPT$("-
",28):"...MONATS - ABSCHLUSS
":RPT$("-",28)
1930 DISPLAY AT(7,1):"MONAT
(IN ZIFFER):" :: ACCEPT AT(7
,21)BEEP VALIDATE(DIGIT)SIZE
(2):MO
1940 OPEN #MO:"DSK1.HAUS/M"&
STR$(MO),OUTPUT,INTERNAL,FIX
ED 255
1950 PRINT #MO:SE,SU
1960 FOR I=1 TO 20
1970 PRINT #MO:SB(I),
1980 NEXT I
1990 FOR E=1 TO 8
2000 PRINT #MO:SZ(E),
2010 NEXT E
2020 CLOSE #MO
2030 SE,SU,U=0 :: FOR I=1 TO
 20 :: SB(I)=0 :: NEXT I ::
FOR E=1 TO 8 :: SZ(E)=0 :: N
EXT E :: GOTO 1780
2040 CALL CLEAR :: END
2050 ! AUSDRUCKEN
2060 GZ,GS=0
2070 OPEN #2:"PIO.LF"
2080 PRINT #2:CHR$(27);CHR$(
64);
2090 PRINT #2:CHR$(27);"1";C
HR$(8);
2100 PRINT #2:CHR$(14);
```

»

```
2110 PRINT #2:"AUSWERTUNG VO
M "&DATUM$
2120 PRINT #2:"":""
2130 PRINT #2:CHR$(14);"AUSG
ABEN" :: RESTORE 1500 :: PRI
NT #2:""
2140 IMAGE #######.##
2150 FOR F=1 TO 20 :: READ A
USG$ :: PRINT #2:AUSG$,,:: P
RINT #2,USING 2140:SB(F):: G
S=GS+SB(F):: NEXT F :: PRINT
 #2:""
2160 PRINT #2:"SUMME",,:: PR
INT #2,USING 2140:GS
2170 PRINT #2:"REAL:",,:: PR
INT #2,USING 2140:SU
2180 PRINT #2:"":""
2190 PRINT #2:CHR$(14);"EINN
AHMEN" :: RESTORE 1620 :: PR
INT #2:""
2200 FOR E=1 TO 8 :: READ EI
NN$ :: PRINT #2:EINN$,,:: PR
INT #2,USING 2140:SZ(E):: GZ
=GZ+SZ(E):: NEXT E :: PRINT
#2:""
2210 PRINT #2:"SUMME",,:: PR
INT #2,USING 2140:GZ
2220 PRINT #2:"REAL:",,:: PR
INT #2,USING 2140:SE
2230 PRINT #2:"":"":""
2240 PRINT #2:CHR$(27);"-";C
HR$(1);
2250 IF SE>SU THEN PRINT #2:
"EINNAHMEN-PLUS:..";SE-SU;"D
M" ELSE PRINT #2:"AUSGABEN Z
U HOCH UM";SU-SE;"DM"
2260 PRINT #2:CHR$(27);"-";C
HR$(0)
2270 PRINT #2:"":"":"" :: PR
INT #2:CHR$(14);"KONTOSTAND"
 :: PRINT #2:""
2280 PRINT #2:"KASSE / BAR",
,:: PRINT #2,USING 2140:KS
2290 PRINT #2:"GIROKONTO  ",
,:: PRINT #2,USING 2140:KG
2300 PRINT #2:"POSTSCHECKKON
TO",:: PRINT #2,USING 2140:K
P
2310 PRINT #2:"SPARKONTO..",
,:: PRINT #2,USING 2140:KZ
2320 PRINT #2:"KREDITKONTO",
,:: PRINT #2,USING 2140:KK
2330 PRINT #2:""
2340 IF (KS+KG+KP+KZ+KK)>0 T
HEN 2350 ELSE 2360
2350 PRINT #2:"GUTHABEN",,::
 PRINT #2,USING 2140:KS+KG+K
P+KK+KZ :: GOTO 2370
2360 PRINT #2:"SCHULDEN",,::
 PRINT #2,USING 2140:ABS(KG+
KP+KK)
2370 PRINT #2:CHR$(12);:: CL
OSE #2
2380 GOTO 320
```

```
100 ! *********************
110 ! *                   *
120 ! *     Haus/Gen      *
130 ! *   Copyright by    *
140 ! *    A.Nitschke     *
150 ! *                   *
160 ! *  Bearbeitet  vom  *
170 ! *     Team des      *
180 ! *  Aktuell Verlages *
190 ! *     Muenchen      *
200 ! *                   *
210 ! * Benoetigte Geraete *
220 ! *  TI99/4A  Konsole *
230 ! *     Ext. Basic    *
240 ! *  Diskettenstation *
250 ! *  Programm Haus/Ja *
260 ! *  Programm Haus/Mo *
270 ! *                   *
280 ! *  Speicherbelegung *
290 ! *     399  Bytes    *
300 ! *                   *
310 ! *********************
320 !
330 ! PROGRAMM GENERIERT 12
DATEIEN FÜR PROGRAMM HAUS/JA
340 DIM SB(20),SZ(8)
350 CALL CLEAR
360 FOR MO=1 TO 12
370 DISPLAY AT(10,1):"ICH OE
FFNE DATEI F. MONAT ";STR$(M
O)
380 OPEN #MO:"DSK1.HAUS/M"&S
TR$(MO),OUTPUT,INTERNAL,FIXE
D 255
390 PRINT #MO:SE,SU
400 FOR I=1 TO 20
410 PRINT #MO:SB(I),
420 NEXT I
430 FOR E=1 TO 8
440 PRINT #MO:SZ(E),
450 NEXT E
460 CLOSE #MO
470 NEXT MO
480 END
```

```
100 ! *********************
110 ! *                   *
120 ! *      Haus/Ja      *
130 ! *   Copyright  by   *
140 ! *    A. Nitschke    *
150 ! *                   *
160 ! *   Bearbeitet vom  *
170 ! *      Team des     *
180 ! * Aktuell  Verlages *
190 ! *      Muenchen     *
200 ! *                   *
210 ! * Benoetigte Geraete *
220 ! *  TI99/4a  Konsole *
230 ! *     Ext. Basic    *
240 ! *  Diskettenstation *
250 ! *   moegl. Drucker  *
260 ! *  Programm Haus/mo *
270 ! *  Programm Haus/Gen *
280 ! *                   *
290 ! *  Speicherbelegung *
300 ! *     6503 Bytes    *
310 ! *                   *
320 ! *********************

330 !
340 CALL CHAR(124,"101010101
010101",125,"000000FFFF")
350 ! *  MONATSSUMMEN  *****
360 DIM SB(12,20),SZ(12,8),S
E(12),SU(12),A$(20),MON$(12)
,SAU(20),SEI(20)
370 CALL CLEAR :: CALL SCREE
N(4)
380 DISPLAY AT(3,1):RPT$("*"
,28):"** JAHRESAUSWERTUNG..1
984 **":"** EINNAHMEN UND AU
SGABEN **":RPT$("*",28)
390 DISPLAY AT(11,1):"WAHLLI
STE"
400 DISPLAY AT(15,3):"*1* EI
NLESEN/MONAT":"..*2* AUSWERT
UNG: AUSGABEN":"..*3* AUSWER
TUNG: EINNAHMEN":"..*4* AUSW
ERTUNG: TOTAL"
410 DISPLAY AT(19,3):"*5* AU
SDRUCKEN":"..*6* ENDE" :: DI
SPLAY AT(23,1)BEEP:"KENNUMME
R EINGEBEN"
420 CALL KEY(0,K,STATUS)
430 IF (STATUS=0)+(K<48)+(K>
54)<>0 THEN 420
440 ON K-48 GOTO 460,670,810
,950,1140,1120
450 ! EINLESEN DER MONATSSUM
MEN
460 CALL CLEAR :: DISPLAY AT
(3,1):RPT$("-",28):"* EINLES
EN DER MONATSSUMMEN":RPT$("-
",28)
470 DISPLAY AT(7,1):"VON MON
AT:" :: ACCEPT AT(7,13)VALID
ATE(DIGIT)SIZE(2):NA
480 DISPLAY AT(9,1):"BIS MON
AT:" :: ACCEPT AT(9,13)VALID
ATE(DIGIT)SIZE(2):NZ
490 IF NA>12 OR NA>NZ THEN 5
00 ELSE 510
500 DISPLAY AT(11,1)BEEP:"FE
HLER !" :: GOTO 470
510 DISPLAY AT(7,1):"" :: DI
SPLAY AT(9,1):"" :: DISPLAY
AT(11,1):""
520 DISPLAY AT(11,5):"BITTE
UM GEDULD ..."
530 RESTORE 1630
540 FOR X=1 TO 12 :: READ MO
N$(X):: NEXT X
550 FOR MO=NA TO NZ
560 DISPLAY AT(14,4):"ICH LE
SE MONAT:" :: DISPLAY AT(14,
22):MON$(MO)
570 OPEN #MO:"DSK1.HAUS/M"&S
TR$(MO),INTERNAL,INPUT ,FIXE
D 255
580 INPUT #MO:SE(MO),SU(MO)
590 FOR I=1 TO 20 :: INPUT #
MO:SB(MO,I),:: NEXT I
600 FOR E=1 TO 8 :: INPUT #M
O:SZ(MO,E),:: NEXT E
610 CLOSE #MO
620 NEXT MO
630 GOTO 370
640 DATA HAUSHALT,MIETE/NK,E
INRICHTUNG,KLEIDUNG,AUTO,VER
SICHERUNG,AUSGEHEN,CASINO,TE
LEFON,URLAUB,FS/RADIO
650 DATA ZEITSCHRIFTEN,COMPU
TER,SCHALLPLATTEN,VIDEO,WEIT
ERBILDUNG,TOTO/LOTTO,KREDIT,
UEBERWEISUNG,SONSTIGES
660 DATA GEHALT,TR-GELD,VERS
ICHERUNG,ZINSEN,GUTSCHRIFTEN
,VERKAEUFE,TOTO/LOTTO,SONSTI
GES
670 ! AUSWERTUNG AUSGABEN
680 CALL CLEAR :: RESTORE 64
0 :: FOR I=1 TO 20 :: READ A
$(I):: DISPLAY AT(I,1):USING
 "##":I :: DISPLAY AT(I,5)SI
ZE(8):A$(I):: NEXT I
```

```
690 CALL VCHAR(1,16,124,20)
700 DISPLAY AT(23,1):"AUSG.A
RT: 0" :: ACCEPT AT(23,11)BE
EP VALIDATE(DIGIT)SIZE(-2):A
 :: IF A=0 THEN 370 :: IF A<
1 OR A>20 THEN 700
710 DISPLAY AT(23,1):""
720 DISPLAY AT(1,16):A$(A)::
 DISPLAY AT(2,16):RPT$("-",L
EN(A$(A)))
730 SSAU,SXAU=0
740 RESTORE 1630
750 FOR MM=1 TO 12 :: READ M
ONAT$ :: DISPLAY AT(MM+6,16)
:MONAT$ :: DISPLAY AT(MM+6,2
1):USING "#####.##":SB(MM,A)
:: SSAU=SSAU+SB(MM,A):: NEXT
 MM
760 CALL HCHAR(20,17,125,14)
770 DISPLAY AT(22,15):"SUM"
:: DISPLAY AT(22,20):USING "
######.##":SSAU
780 SXAU=SSAU/(NZ+1-NA):: DI
SPLAY AT(24,15):"1/"&STR$(NZ
+1-NA):: DISPLAY AT(24,21):U
SING "#####.##":SXAU
790 GOTO 700
800 !
810 ! AUSWERTUNG EINNAHMEN
820 !
830 CALL CLEAR :: RESTORE 66
0 :: FOR I=1 TO 8 :: READ A$
(I):: DISPLAY AT(I+5,1):USIN
G "##":I :: DISPLAY AT(I+5,5
)SIZE(8):A$(I):: NEXT I
840 CALL VCHAR(1,16,124,20)
850 DISPLAY AT(23,1):"EINN.A
RT: 0" :: ACCEPT AT(23,11)BE
EP VALIDATE(DIGIT)SIZE(-1):A
 :: IF A=0 THEN 370 :: IF A<
1 OR A>8 THEN 850
860 DISPLAY AT(23,1):""
870 DISPLAY AT(1,16):A$(A)::
 DISPLAY AT(2,16):RPT$("-",L
EN(A$(A)))
880 SSEI,SXEI=0
890 RESTORE 1630
900 FOR MM=1 TO 12 :: READ M
ONAT$ :: DISPLAY AT(MM+6,16)
:MONAT$ :: DISPLAY AT(MM+6,2
1):USING "#####.##":SZ(MM,A)
:: SSEI=SSEI+SZ(MM,A):: NEXT
 MM
910 CALL HCHAR(20,17,125,14)
920 DISPLAY AT(22,15):"SUM"
:: DISPLAY AT(22,20):USING "
######.##":SSEI
930 SXEI=SSEI/(NZ+1-NA):: DI
SPLAY AT(24,15):"1/"&STR$(NZ
+1-NA):: DISPLAY AT(24,21):U
SING "#####.##":SXEI
940 GOTO 850
950 CALL CLEAR
960 SSE,SSA,DIF,SXE,SXA,DIX=
0
970 ON ERROR 1000
980 IF NA<>NZ THEN 990 :: DI
SPLAY AT(1,4):"AUSWERTUNG MO
NAT" :: DISPLAY AT(1,23):MON
$(MO-1):: GOTO 1000
990 DISPLAY AT(1,1):"AUSWERT
UNG VON ";MON$(NA);" BIS ";M
ON$(NZ)
1000 DISPLAY AT(4,1):"MON...
.EINNA...AUSGA....DIFF"
1010 CALL HCHAR(5,3,125,28)
1020 RESTORE 1630 :: FOR MM=
1 TO 12 :: READ MONAT$ :: DI
SPLAY AT(MM+6,1):MONAT$
1030 SSE=SSE+SE(MM):: SSA=SS
A+SU(MM):: DIF=SE(MM)-SU(MM)
1040 DISPLAY AT(MM+6,8):USIN
G "#####":SE(MM):: DISPLAY A
T(MM+6,16):USING "#####":SU(
MM)
1050 DISPLAY AT(MM+6,24):USI
NG "#####":DIF
1060 NEXT MM
1070 CALL HCHAR(20,3,125,28)
1080 DISPLAY AT(22,1):"SUM"
:: DISPLAY AT(22,7):USING "#
#####":SSE :: DISPLAY AT(22,
15):USING "######":SSA :: DI
SPLAY AT(22,24):USING "#####
":SSE-SSA
1090 DISPLAY AT(24,1):"1/"&S
TR$(NZ+1-NA):: DISPLAY AT(24
,7):USING "######":SSE/(NZ+1
-NA):: DISPLAY AT(24,15):USI
NG "######":SSA/(NZ+1-NA)
1100 DISPLAY AT(24,24):USING
 "#####":(SSE-SSA)/(NZ+1-NA)
1110 CALL KEY(0,K,STATUS)::
IF STATUS=0 THEN 1110 :: GOT
O 370
1120 CALL CLEAR :: END
1130 ! AUSDRUCKEN **********
&
1140 OPEN #20:"PIO.LF"
```

```
1150 PRINT #20:CHR$(27);CHR$
(64);:: PRINT #20:CHR$(27);"
1";CHR$(3);
1160 PRINT #20:CHR$(14);"JAH
RESÜBERSICHT..1984": : : : :
1170 PRINT #20:CHR$(14);"AUS
GABEN": : : :
1180 IMAGE #######
1190 IMAGE #########
1200 PRINT #20:CHR$(27);"-";
CHR$(1);
1210 PRINT #20:"MONAT...HAUS
H..MIETE..EINRI..KLEID...AUT
O..VERSI..AUSGE..CASIN..TELE
F..URLAU" :: PRINT #20:CHR$(
27);"-";CHR$(0);
1220 RESTORE 1630 :: PRINT #
20
1230 FOR M=1 TO 12 :: READ M
ON$(M):: NEXT M
1240 FOR MM=NA TO NZ :: PRIN
T #20:MON$(MM);"...";:: FOR
A=1 TO 10 :: PRINT #20,USING
 1180:SB(MM,A);:: SAU(A)=SAU
(A)+SB(MM,A)
1250 NEXT A :: PRINT #20 ::
NEXT MM
1260 PRINT #20 :: PRINT #20:
"SUMME ";:: FOR A=1 TO 10 ::
 PRINT #20,USING 1180:SAU(A)
;:: NEXT A
1270 PRINT #20:"SCHN. ";:: F
OR A=1 TO 10 :: PRINT #20,US
ING 1180:SAU(A)/(NZ+1-NA);::
 NEXT A :: PRINT #20:"":"":"
"
1280 PRINT #20:CHR$(27);"-";
CHR$(1);
1290 PRINT #20:"MONAT...RADI
O...ZEIT..COMPU..SCHAL..VIDE
O..W.BIL..LOTTO..KREDT..ÜBER
W..SONST" :: PRINT #20:CHR$(
27);"-";CHR$(0);
1300 RESTORE 1630 :: PRINT #
20
1310 FOR M=1 TO 12 :: READ M
ON$(M):: NEXT M
1320 FOR MM=NA TO NZ :: PRIN
T #20:MON$(MM);"    ";:: FOR
A=11 TO 20 :: PRINT #20,USIN
G 1180:SB(MM,A);:: SAU(A)=SA
U(A)+SB(MM,A)
1330 NEXT A :: PRINT #20 ::
NEXT MM
1340 PRINT #20 :: PRINT #20:
"SUMME ";:: FOR A=11 TO 20 :
: PRINT #20,USING 1180:SAU(A
);:: NEXT A
1350 PRINT #20:"SCHN. ";:: F
OR A=11 TO 20 :: PRINT #20,U
SING 1180:SAU(A)/(NZ+1-NA);:
: NEXT A :: PRINT #20:"":"":
"":"":""
1360 IF (NZ+1-NA)>10 THEN PR
INT #20:CHR$(12);
1370 PRINT #20:CHR$(14);"EIN
NAHMEN": : : :
1380 PRINT #20:CHR$(27);"-";
CHR$(1);
1390 PRINT #20:"MONAT...GEHA
LT....TR-GE....VERSI.....ZIN
S....GUTSC....VERKA....LOTTO
....SONST" :: PRINT #20:CHR$
(27);"-";CHR$(0);
1400 PRINT #20
1410 FOR MM=NA TO NZ :: PRIN
T #20:MON$(MM);"..";:: FOR I
=1 TO 8 :: PRINT #20,USING 1
190:SZ(MM,I);:: SEI(I)=SEI(I
)+SZ(MM,I)
1420 NEXT I :: PRINT #20 ::
NEXT MM
1430 PRINT #20 :: PRINT #20:
"SUMME";:: FOR I=1 TO 8 :: P
RINT #20,USING 1190:SEI(I);:
: NEXT I
1440 PRINT #20:"SCHN.";:: FO
R I=1 TO 8 :: PRINT #20,USIN
G 1190:SEI(I)/(NZ+1-NA);:: N
EXT I :: PRINT #20:"":"":"":
"":""
1450 IF (NZ+1-NA)<5 OR(NZ+1-
NA)>10 THEN 1470
1460 PRINT #20:CHR$(12);
1470 PRINT #20:CHR$(14);"GES
AMT-ÜBERSICHT": : : :
1480 PRINT #20:CHR$(27);"-";
CHR$(1);
1490 PRINT #20:"MONAT.......
......EINNAHMEN..........AUS
GABEN.........DIFFERENZ" ::
PRINT #20:CHR$(27);"-";CHR$(
0);
1500 PRINT #20
1510 FOR MM=NA TO NZ :: PRIN
T #20:MON$(MM);:: PRINT #20:
RPT$(" ",15);:: PRINT #20,US
ING 1190:SE(MM);:: PRINT #20
:RPT$(" ",9);:: GEI=GEI+SE(MM)
```

```
1520 PRINT #20,USING 1190:SU
(MM);:: PRINT #20:RPT$(" ",9
);:: GAU=GAU+SU(MM)
1530 PRINT #20,USING 1190:SE
(MM)-SU(MM):: NEXT MM :: PRI
NT #20
1540 PRINT #20:"SUMME";:: PR
INT #20:RPT$(" ",13);:: PRIN
T #20,USING 1190:GEI;
1550 PRINT #20:RPT$(" ",9);:
: PRINT #20,USING 1190:GAU;:
: PRINT #20:RPT$(" ",9);
1560 PRINT #20,USING 1190:GE
I-GAU
1570 PRINT #20:"SCHN.";:: PR
INT #20:RPT$(" ",13);:: PRIN
T #20,USING 1190:GEI/(NZ+1-N
A);
1580 PRINT #20:RPT$(" ",9);:
: PRINT #20,USING 1190:GAU/(
NZ+1-NA);:: PRINT #20:RPT$("
 ",9);
1590 PRINT #20,USING 1190:(G
EI-GAU)/(NZ+1-NA)
1600 PRINT #20:CHR$(12);
1610 PRINT #20:CHR$(27);CHR$
(64);:: CLOSE #20
1620 RUN
1630 DATA JAN,FEB,MAR,APR,MA
Y,JUN,JUL,AUG,SEP,OCT,NOV,DE
C
```

HAUS/JA

Haus/Ja ist die Erweiterung des Programmes Haus/Mo. Mit diesem Programm können Sie die abgeschlossenen Monatsdateien aus Programm Haus/Mo einlesen und auswerten. Zuerst müssen Sie jedoch das Programm Haus/Gen laden, welches Ihnen die für dieses Programm erforderlichen Dateien eröffnet. Danach Haus/Gen wieder löschen und Haus/Ja laden. Jetzt haben Sie die Auswahl, ob Sie einen Monat, mehrere Monate oder das gesamte Jahr ausgewertet haben wollen. Der Computer holt sich nun die für die Auswertung erforderlichen Dateien und gibt Ihnen in Tabellenform eine genaue Aufstellung Ihrer Geldbewegungen für den gewünschten Zeitraum, sowie den monatlichen Durchschnitt. Die Auswertung kann sowohl für einzelne Posten, als auch für die monatlichen Gesamtsummen gewählt werden.

Alfred Nitschke

HAUS/GEN

Haus/Gen eröffnet die für das Programm Haus/Gen erforderlichen Dateien.
Bevor Sie das Programm Haus/Gen benutzen, bitte zuerst dieses Programm laden. Der Computer informiert Sie über den Bildschirm, welche Datei er gerade eröffnet. Sind alle Dateien eröffnet, ist dieses Programm wieder aus dem Speicher zu löschen und Programm Haus/Ja zu laden. Danach können Sie mit der Auswertung, wie in Programm Haus/Ja beschrieben, beginnen.

Alfred Nitschke

TI-PRIM

Dieses Programm können Sie zur Ermittlung der Primzahlen zwischen 1 und einer beliebigen Zahl unterhalb von 1000 verwenden. Die Höchstgrenze von 1000 ist deshalb gewählt, weil der Computer die zu untersuchenden Zahlen zuvor in eine Tabelle einliest. Bei höherer Dimensionierung würde das System abstürzen. Nach Start des Programmes gibt der Computer in der Kopfzeile den Anfangswert '1' aus und Sie wählen nun die gewünschte Höchstgrenze. Nach bBeendigung des Rechenvorganges erscheint in der Kopfzeile die Menge der gefundenen Primzahlen. Kurz darauf werden die Primzahlen in Tabellenform ausgegeben. Sollte der Computer mehr als 48 Zahlen gefunden haben, unterbricht er die Ausgabe nach der 48. Zahl, da der vorhandene Bildschirmplatz jetzt gefüllt ist. Durch Betätigen von ENTER werden diese Primzahlen gelöscht und die nächsten 48er Folge ausgegeben.
Am Ende des Programmes können Sie wählen, ob Sie das Programm verlassen wollen oder eine neue Tabelle wünschen. Geben Sie hierzu ein 'J' ein.

Hartmut König

```
100 ! **********************
110 ! *                    *
120 ! *  TI-PRIM           *
130 ! *    Copyright by    *
140 ! *   Hartmut Koenig   *
150 ! *                    *
160 ! *   Bearbeitet vom   *
170 ! *      Team des      *
180 ! *  Aktuell Verlages  *
190 ! *      Muenchen      *
200 ! *                    *
210 ! * Benoetigte Geraete *
220 ! *  TI99/4A  Konsole  *
230 ! *     Ext. Basic     *
240 ! *                    *
250 ! *  Speicherbelegung  *
260 ! *     955  Bytes     *
270 ! **********************
280 !
290 DIM HK(1000)
300 ON WARNING NEXT :: ON BR
EAK NEXT
310 CALL CHAR(95,"000000FFFF
"):: CALL SCREEN(5):: FOR I=
1 TO 12 :: CALL COLOR(I,2,8)
:: NEXT I :: CALL COLOR(14,5
,5)
320 CALL CLEAR :: DISPLAY AT
(3,9):"PRIMZAHLEN" :: DISPLA
Y AT(4,9):"__________" :: DI
SPLAY AT(7,2):"ANFANGSWERT M
ENGE ENDWERT"
330 DISPLAY AT(8,2):"_______
____ _____ _______" :: DISPL
AY AT(10,6):"01       00
 ???"
```

```
340 ACCEPT AT(10,22)VALIDATE
(DIGIT):B :: DISPLAY AT(12,6
):"MOMENT BITTE..." :: FOR I
=1 TO B :: HK(I)=I :: NEXT I
350 FOR I=2 TO INT(SQR(B))::
 IF HK(I)=0 THEN 370
360 FOR K=I*2 TO B STEP I ::
 HK(K)=0 :: NEXT K
370 NEXT I
380 J=0 :: FOR N=1 TO B :: I
F HK(N)<>0 THEN J=J+1
390 NEXT N :: DISPLAY AT(10,
14)SIZE(-3):USING "###":J
400 !  *TABELLE VERDICHTEN*
410 L=1 :: FOR K=1 TO B :: I
F HK(K)=0 THEN 420 ELSE HK(L
)=HK(K):: L=L+1
420 NEXT K
430 Z=12 :: SP=2 :: FOR N=1
TO J :: DISPLAY AT(Z,SP):USI
NG "###":HK(N):: SP=SP+4 ::
IF SP>26 THEN SP=2 :: Z=Z+2
440 IF Z=22 AND SP>25 THEN 4
50 ELSE 460
450 DISPLAY AT(24,5):"WEITER
 MIT  <ENTER>" :: CALL KEY(0
,K,S):: IF S=0 THEN 450 ELSE
 Z=12 :: SP=2 :: CALL HCHAR(
12,1,32,416)
460 NEXT N
470 DISPLAY AT(24,7):"NOCHMA
L ?J" :: ACCEPT AT(24,16)SIZ
E(-1)VALIDATE("JNjn"):AN$ ::
 IF AN$="J" OR AN$="j" THEN
320 ELSE CALL CLEAR :: STOP
```

BÖRSE

TI GRAFIK ADVENTURE Generator Set 15 DM Info und Liste gegen Rückporto bei Thomas Schröter, Mönchgraben 26, 4 Düsseldorf 13

VERK. Expansions-Box mit Karte für Box, Controller-Karte, Laufwerk, Disk-Manager 2 — Modul, 19 Disketten mit Spitzenprogramm. Module: Parsec, Schach, Speech-editor, early learning ... etc. Sprachsynthesizer, Literatur, Prog. für Disk und gute x Basic. Programme gegen Angebot zusammen oder einzeln. Klaus Versin, Heerstr. 377 5014 Kerpen-Brüggen.

R+B SOFT/R+B SOFT/ R+B SOFT Hex Dezimal Umrech. Spreit-Edt Telefon-Datei mit schnellstmög. Cass-Speich / je 12 DM + NN / Wir erstellen für Sie Programme und geben Tips / Tel.: 04936/ 64 76 / 29 87 Grossheide 1, Kiebitzweg 4.

TAUSCHE TI-Module Advent. + 10 Kass. + Parsec + Statistik + Rechenkünst. gegen RS 232 Karte oer anderes auch einzeln. Tel. 06408/76 50 — 20 — 22 Uhr.

SUCHE für TI 99/4A EX-Basic + DSK 1 + 32 K) Programm f. Hausverw. + Nebenkostenabrechn. Angebot an G. Kosel, 6057 Dietzenbach, Rotdornweg 22

TI 99/4A: Tausche Modul „Buck-Rogers" Neupr. 149 DM gegen Adventur-Modul + Spiele oder gegen gleichw. Module! Angeb. an Frank Kappler, Robert-Kochstr. 27, 6652 Bexbach

SUCHE für TI 99/4A Ext. Basic RS 232 Diskettenlaufwerk und Drucker. Kurt Meier, Delsenbachweg 38, 8500 Nürnberg 90 Tel. 0911/39 63 30 ab 17.00 Uhr

SUCHE für TI 99/4A Interface zum Druckeranschluß (Seikosha GP-100 A Mark II. (ggf. auch beide Teile zusammen.) Lothar Best, Lessingstr. 34, 6473 Cedern 1, Tel. 06045/12 79

TI 99/4A suche User im Raum Euskirchen-Rheinbach-Mechernich. Ich möchte Assembler lernen u. Basic vertiefen. Suche 32 K erw. R. Breuer, Schmuckstr. 6, 5358 Bad Muenstereifel-Soller 02257/571 nach 18.00

TI 99/4A Software TI — und Extended Basic. Anwenderprogramme und Spiele. Info gegen Rückporto bei: JHC-Software; Chattenstr. 28, 6331 Katzenfurt.

VERK. BÜCHER mit Progr. z.B. Superspiele F.D. TI usw. Schallpl.-Archiv, Basic F. Anfänger, alles gebr. Alles halb. Pr. 06638/15 03

STATIK-PROGRAMME FÜR TI 99/4A Kehlbalkendach, Fachwerk, Stabwerk Durchlaufträger mit Bemessung kreuzw. bew. Platten bel. Belastung alle Programme mit Textverarb. Ing.-Büro B. Rath 4790 Paderborn, Milchstr. I

VERKAUFE: Origin. Module (Videospiele I+II, CAR Wars, Fittness-Trainer) a 40 DM & Orig. Kassetten (Finanzberater, Basic-XBasiskurs) a 15 DM. Tel. 06471/528 93

TI 99/4A 250,— X-Basic 250,— 32 Kram zum einbauen in die Konsole 600,— Dat. u. anal. 115,— Dat. u. Text 115,— Statistik 70,— Viele Mod. Bocher Tel. Zang 040/ 722 55 01 84 ab 17 Uhr 650 10 74

TI 99/4A — biete Schach Modul 80 DM Basic Kurs 15 DM Ex. Basic Kurs 15,— DM. Tel. 06262/ 29 16

COUPON

Die große Börse für jeden Zweck in TI REVUE. Wenn Sie etwas suchen, etwas verkaufen oder tauschen wollen, Kontakte wünschen oder, oder, oder, dann ist TI REVUE die richtige Zeitschrift für Sie. Zum sagenhaften Kennenlern-Preis von DM 1,50 pro Zeile! Einfach ausfüllen, ausschneiden und zusammen mit einem Verrechnungsscheck über die entsprechende Summe (oder in Briefmarken) absenden an

2 Zeilen	3,00 DM
4 Zeilen	6,00 DM
6 Zeilen	9,00 DM
8 Zeilen	12,00 DM
10 Zeilen	15,00 DM

TI REVUE
Anzeigenabtlg.
Postfach 1107
8044 Lohhof

() Bitte hier ankreuzen, wenn die Anzeige unter Chiffre erscbeinen soll. Chiffre-Gebühr 10 DM.

Hiermit bestelle ich obige Kleinanzeige. Einen Verrecbnungsscheck über DM ________ habe ich beigelegt. Mir ist bekannt, daß Anzeigenaufträge obne ausgefüllten und unterschriebenen Verrechnungsscheck nicht bearbeitet werden.

Name ____________ Vorname ____________

Strasse ____________ (PLZ) Ort ____________

BÖRSE

TI 99/4A + Ext. Basic + EP 22 + V24 Int. + Rec. + Kabel + Div. Bücher + 50 Kass. + Joyst. Adapter + Color-TV + Jede Menge Softw. F. VHB 1850,–. Arno Zimmermann, PF. 122, 6087 Büttelborn 1

VERK. TI 99/4A + Ex. Basic + Handb. + Rec. + Kabel + Joyst + Litt Basic + Ex Basic Lerncass + Progr / 9 Mon. alt. FP 800 DM / H. Ryll, Tel. 02359/77 76

TI 99/4A – X – Bas. Modul 150,– Datenverw. + Analyse 50,– Beg. Grammar 30,– Ext. Centronics Interface direkt an Konsole 150,– (z.B. Seik. GP – 80 A) Mattel Intellivision mit 6 Spielkas. 200,–Buchungsjournal DM 190,– minus Mission DM 25,– Bernhardt, Domb. Sande R 133, 506 Berg. Gladba. 02202/381 00

TI – 99/4A: Suche gut erhaltenen Drucker mit Schnittstelle sowie Peri-Box und Diskettenlaufwerk: Wolfg. Schmitz, (02101/49 54 41)

SUCHE Briefkontakt zu TI-Besitzern im Raume Ruhrgebiet zwecks Erfahrungsaustausches.
G. Michel, Duissernstr. 33, 4100 Duisburg. Tel. 0203/33 32 34

PHP 1900 THERMO-DRUCKER zu verkaufen 1 Monat alt VB 700,– DM direkt an Konsole anschließbar NP. 1400, 30 Zeichen / S.A. Krause, Lauterbacherstr. 11, 6-FFM 61, 069/41 16 77

DRUCKE Ihre Listings von Diskette oder Cassette. TI-Basis/Ext. Basic 100 DM + NH oder 100 DM-Schein + 2,50 Briefmarke. Peter Voigt, Breslauerstr. 20, 4060 Viersen 11

SUCHE TI Extended BASIC Tel. 0531/892 66 22

EXT. SPITZENSPIEL! endlich eins wie in der Hölle bei begrü. Nichtgefallen Geld zurück!! best. kein Schrott. Cass. für 20 DM bei Gripp, Gustav-Poensgen 33, 4-Düsseldorf

TI 99/4A PROGRAMME Info gegen 2 DM in Briefmarken Andreas Schmidt am Storchenbaum 9, 6793 Bruchmühlbach-Miesau

SUCHE externe RS 232 Box für TI 99/4A ab 18 Uhr Tel. 040/81 23 16

TI 99/4A EXT. B. RS 232 Speech, Progs. Lit. NP. 1500,– VB 950,– Tel. 09133/12 29

FAWCETT-SOFTWARE präsentiert Englische Programme in deutsch. Liste anfordern. Martin Zeddies, Ligusterweg 3, 3180 Wolfsburg 11

SUCHE DT. BEDIENUNGSANLEITUNG für Epsondrucker MX 80 Typ 1. Eine Kopie ist auch erwünscht. H.J. Bikkel, Vockrathserstr. 50, 4040 Neuss 22, 02101/84620 ab 17.00 Uhr

US TI 44/4A zu verkaufen incl. Handbücher und Lektionscass. engl. Angebot an 06021/601 16. Versand nur gegen NN oder V-Scheck

TI – 99/4A: Verk. PRG. Routinen + Lagerverwalt. zus. DM 195,– auch einzeln Tel. 0251/664 84

VERKAUFE TI 99/4A + Ext. Basic + Cassettenrec.-Kabel (Orig. TI) + TI Bücher (TI PS & Tricks, Handbuch TI/Ex. Basic Etc.) Für 550,– DM. Tel. 0821/227 41/Abend

SUCHE Kontakt mit TI Besitzern in Augsburg, Jörg Storm, Christleseeweg 34, Tel. 0821/638 57

SUCHE Disassembler, Speicheradressen von Konsolenroutinen, Programmtausch und Erfahrungstausch mit Assemblerfans. M. Humbert, Saphirring 30, 3008 Garbsen 1.

ACHTUNG! Gelegenheit: Große Menge origin. verpackte Adventure-Games + Module v. Texas verk. nur Tel! ab 16 Uhr 0841/4 40 00. Achtung

SUCHE: Terminal Emulator II RS 232 extern, Konsolenschaltungsunterlagen, A/D Interface.
Dirk Fortmann, Wertherstr., 4448 Bielefeld 1, Tel. 12 38 71

SUCHE für TI-99 verschiedene Hardware wie Laufwerke, Module, Mini-Memory, Peri-Box, 32 K-Karte usw. auch gebraucht od. defekt. Suche auch gute Software. Maier Heinz, Hauptstr. 24, 8077 Reichertshofen 2, Tel. ab 16.15 Uhr 08453/25 85.

ACHTUNG! Verkaufe verschiedene Module (z.B. Munch Man) für 40,– DM das Stück. Verkaufe E-PROMS 2716 solange Vorrat reicht für 6,–. SUCHE deutsches Handbuch für Mini Memory, Diskkontroller extern sowie Speichererweiterung extern. Reinhard Weinhold, Vor dem Buchwald 8. Tel. 06426/7716 nach 17.00 Uhr.

VERK. neuwert. TI 99/4A Module Munch-Man: 70,–/Yatzee: 30, Alle 4 Module 140,– ... Suche Mini Memory. Andreas Braun, Ebereschenweg 1, 5107 Simmerath.

TI99/4A+EXT. BASIC BESITZER Ich tausche oder verkaufe eine Auswahl von 18 Programmen aus meiner Sammlung. Alles in Ext. Basic geschrieben und alles hervorragende Grafiken und Ideen.
Ruf mich an – meistens bin ich erst abends zu erreichen: H. Dirks, Tel. 0241 – 87 22 05

SUCHE defekte Joystick's und Kontakte für TI-99/4A. Klaus Schau, Roderbruchstr. 32, 3 Hannover 51

FORTH-Assemblerlistings für alle wichtigen Micro-Prozessoren je 25 DM: Installation-Manual 25 DM Autolinker u. Buchhaltung f. Appel 11 kostenl. Info anf. bei B. Lipgens-Datentechnik, Wilbrechtstr. 65, 8000 München 71

VERKAUFE Original TI-Laufwerk PHP 1250 zum Einbau in Peri-Box, 1 Jahr alt, wenig gelaufen. VH 450,–. Suche Kontakte zu anderen TI-99 Anwendern im süddeutschen Raum sowie Usern mit Akustikkopplern. Telefon 07161/3 15 21.

Wer im Raum LÜNEBURG hat Interesse an einem TI-CLUB Jörg Podlinksi, Grenzstr. 6, 2120 Lüneburg / 04131-5 54 57

HYPER OLYMPIC und 230 weitere Spiele in TI u. X-Basic. Info 1 DM bei Frajo Frey, Bergknappenstr. 178, 4350 Recklinghausen.

TI 994 A – 1 PAAR UNGEBR. Joysticks VHB 40 DM, C 64 Suche immer noch Programme und evtl. Gebr. Floppy xxx C 64 Programmtausch. Reiner Keller, 1 BLN 26, Quickb. Str. 91